Building A Quiz Service With Spring Boot

An Introduction To Natural Language Processing

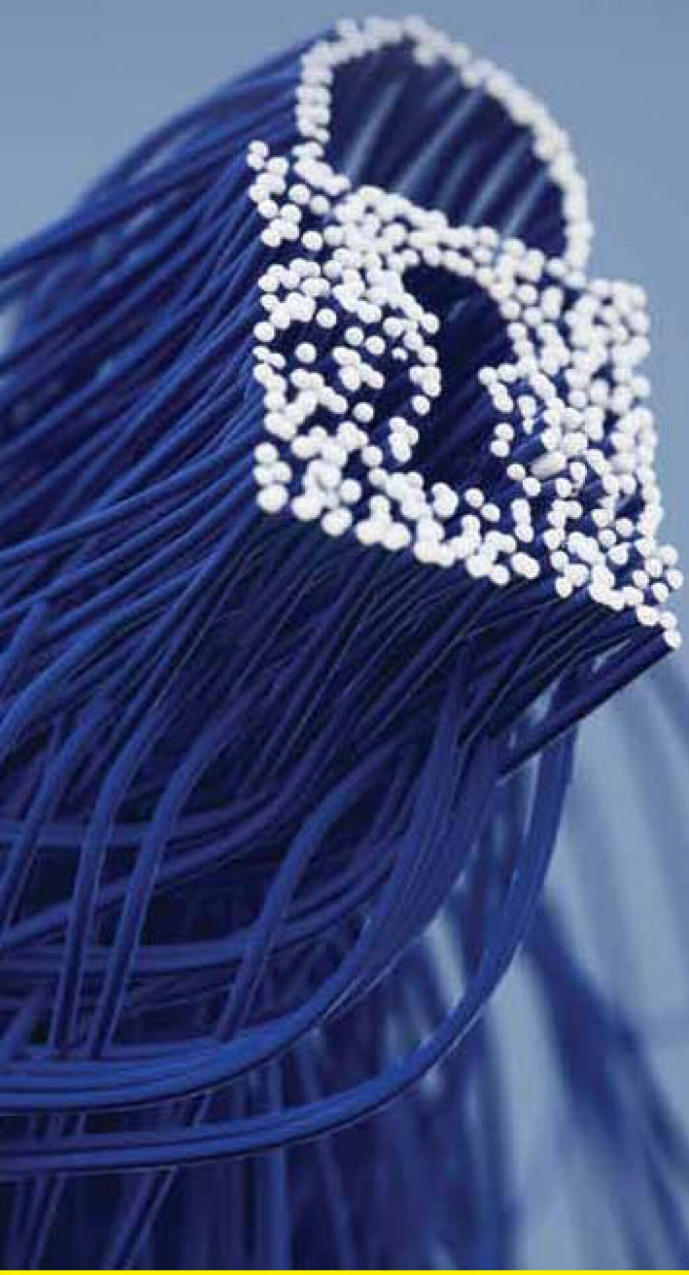
₹120
ISSN-2456-4885

# OpenSource ForYou

THE COMPLETE MAGAZINE ON OPEN SOURCE

Volume: 11 | Issue: 05 | Pages: 100 | March 2023

An EFY GROUP Publication



# Secure Your Applications

Dynamic Application Security Testing Using Acunetix And GuardRails

Static Application Security Testing With SonarQube

AI Tools That Enhance Cloud Security

How To Prevent Cookies From Being Hijacked

Convert ChatGPT Into An Advanced Voice Assistant

# Does Your Antivirus Solution Provide You With Complete Protection?

## Don't RELY on 20 years old technology to FIGHT current date viruses

TOP 10 things that your Antivirus solution should provide to tackle today's threat...

| | | | |
|---|---|---|---|
| Ransomware File Protection | ATP- Advance Threat Protection | EDR- Endpoint Detection And Response | |
| Application Blocking | Deep Learning Malware Analysis | Exploit Prevention | URL Blocking |
| Disk and Boot Record Protection | Peripheral Control | Respond Investigate Remove (Root Cause Analysis) | |

If it doesn't, contact us for a solution which can

Contact: Santosh on 9971696319 or Email at santosh.gupta@itsipl.com

ITS
A Total Solutions Company

**I. T. Solutions India Private Limited**
D-88/5, Okhla Industrial Area, Okhla Phase I, New Delhi -110020
Ph: 011-47695000 • Email: sales@ itsipl.com • www.itsipl.com

- Mumbai
- Jaipur
- Chandigarh

SOPHOS 2020 BEST NEXTGEN PARTNER | Delhi/ NCR
SOPHOS 2019 BEST NEXTGEN PARTNER | Delhi/ NCR
SOPHOS 2018 BEST SI PARTNER | Delhi

# CONTENTS

MARCH 2023 | ISSN-2456-4885

## FOR U & ME



## FOCUS



## DEVELOPERS



## ADMIN










## REGULAR FEATURES

EDITOR
RAHUL CHOPRA

EDITORIAL, SUBSCRIPTIONS & ADVERTISING
Delhi (HQ)
D-87/1, Okhla Industrial Area, Phase I, New Delhi 110020
Phone: +91-9811155335
E-mail: info@efy.in

MISSING ISSUES
Phone: +91-9811155335
E-mail: support@efy.in

BACK ISSUES
Phone: +91-9811155335
E-mail: support@efy.in

NEWSSTAND DISTRIBUTION
Phone: +91-9811155335
E-mail: efycirc@efy.in

ADVERTISEMENTS
NEW DELHI (HEAD OFFICE)
Phone: +91-9811155335
E-mail: efyenq@efy.in

MUMBAI
E-mail: rmowest@efy.in

BENGALURU
E-mail: rmosouth@efy.in

CHINA
Worldwide Focus Media
E-mail: china@efy.in

GERMANY
pms Plantenberg Media Service GmbH
E-mail: germany@efy.in

JAPAN
Tandem Inc.
E-mail: japan@efy.in

TAIWAN
J.K. Media
E-mail: taiwan@efy.in

UNITED KINGDOM
ASA Media
E-mail: uk@efy.in

UNITED STATES
E & Tech Media
E-mail: usa@efy.in

Printed, published and owned by Ramesh Chopra. Printed at Tara Art Printers Pvt Ltd, A-46,47, Sec-5, Noida, on 28th of the previous month, and published from D-87/1, Okhla Industrial Area, Phase I, New Delhi 110020. 


| SUBSCRIPTION RATES | | | |
|---|---|---|---|
| Year | Newstand Price (₹) | You Pay (₹) | Overseas |
| Five | 7200 | 4320 | — |
| Three | 4320 | 3030 | — |
| One | 1440 | 1150 | US$ 120 |

Kindly add ₹ 50/- for outside Delhi cheques.
Please send payments only in favour of EFY Enterprises Pvt Ltd.
Non-receipt of copies may be reported to **support@efy.in**—do mention your subscription number.


# CONTENTS

# Intelligence framework Octosuite is now available on GitHub

Version 3.1.0 of the open source intelligence (OSINT) framework Octosuite has just been made available on GitHub. Octosuite, a Python-based tool, offers a safe and intuitive interface for quickly searching and exploring data pertaining to a repository, organisation, or user. To identify pertinent data fast, it also searches for themes, commits, and issues. Every search result is exported in a CSV file that can be read by other programs.

Users can begin using Octosuite through a command-line interface (CLI) or graphical user interface. While the latter allows users to search commands from a dropdown menu, CLI is more flexible when processing data in batches.

After installing Octosuite, the user must launch it in the terminal. Octosuite will make an effort to establish three directories at launch time — logs for storing session logs, output for saving CSV files, and download for saving source code via the source command.

Since 26 per cent of firms now use open source investigative tools, the market for open source intelligence is anticipated to develop significantly over the next five years.

For open source investigators, security researchers, and anyone who wants to quickly examine and probe data hosted on GitHub, Octosuite is a crucial tool. For instance, it can be used to look into instances like the 2022 GitHub malware attack, in which a single user account compromised more than 35,000 repositories.

## New open source ecosystems will receive up to US$ 28 million from NSF in funding

The National Science Foundation (NSF) in the US is aiming to promote the growth of open source ecosystems in STEM (science, technology, engineering and mathematics) subjects, according to a release. The 'Pathways to Enable Open-Source Ecosystems' or POSE programme will not provide funding for currently operating open source ecosystems, tools, or products but will instead concentrate on assisting fresh open source ecosystems. The POSE programme's objectives, according to the statement, are to increase the number of academics and innovators working on and contributing to open source ecosystems, and to establish risk-free and secure development and contribution channels for high-impact ecosystems.

The estimated budget for NSF's 30 to 50 awards is US$ 27.8 million. Many of the projects that it funds "result in publicly accessible, changeable, and distributable

## Two security flaws emerge with ImageMagick

The open source image processing program ImageMagick has a few security flaws that might possibly result in information exposure or cause a Denial of Service (DoS) event, according to researchers at Metabase Q (CVE-2022-44268, CVE-2022-44267).

Raster and vector picture files can be viewed, converted, and edited using the free and open source software package ImageMagick. When parsing a PNG picture with a file name that only contains a single dash ('-'), the CVE-2022-44267 vulnerability, a DoS problem, can be activated.

When parsing an image, the CVE-2022-44268 vulnerability is an information disclosure bug that can be used to access any files from a server. The software may have included the content of any external file when it parses a PNG image, for example, to resize (if the ImageMagick binary has permissions to read it).

An attacker must use the ImageMagick program to upload a specially created image to a website in order to remotely exploit the flaws. By including a text chunk that specifies certain metadata, such as the file name, which must be set to '-' for exploitation, the attacker can create the picture. The two flaws impact ImageMagick version 7.1.0-49 of the program; they were fixed in version 7.1.0-52, which was released in November 2022.

## Mycroft project shuts down due to lack of funds

The open source, privacy-respecting substitute for Google Home and Amazon Echo, Mycroft, is closing operations due to a lack of funding. Joshua Montgomery, a 15-year business veteran with an aerospace engineering background, founded Mycroft. Montgomery sought to develop a voice assistant that could offer the ease of commercial solutions while protecting user privacy because he has long been an advocate of open source software.

Naturally, the revelation hasn't gone down well with the project's backers, many of whom are wondering why the initiative ran out of money despite receiving funding in many rounds. Others have questioned the company's decision to sell more than 100 machines on eBay for US$ 499 apiece rather than deliver them to backers. Mycroft's tragedy is all too common among crowdsourced hardware projects, with many failing to overcome the financial and logistical challenges of manufacturing.

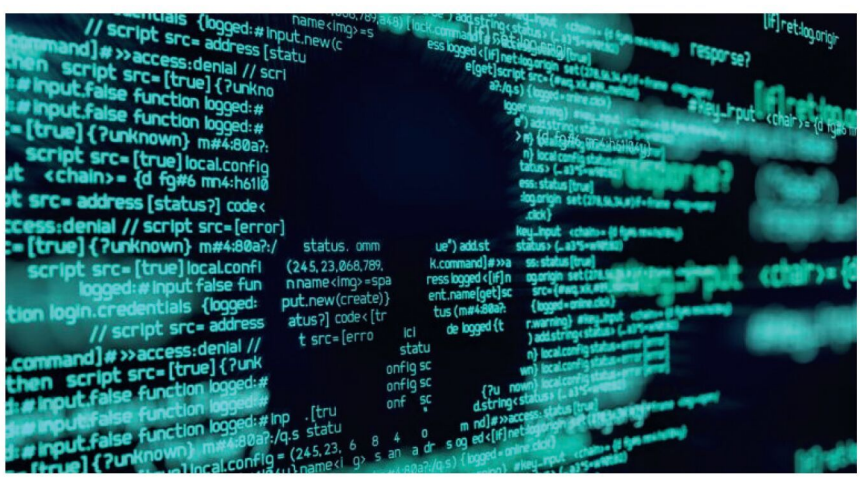
Despite being useful, Amazon Echo and Google Home are a privacy nightmare because they steal enormous amounts of user data. If the Mycroft project is indeed dead, the privacy-conscious will suffer greatly.

open source solutions, including software, hardware, models, specifications, programming languages, or data platforms that spark additional innovation," according to NSF.

The agency's Directorate for Technology, Innovation and Partnerships, in collaboration with the other NSF directorates, developed the POSE initiative, which is accessible for research and innovation by all STEM open source ecosystems. The POSE programme will serve as a "pathway to translate scientific and engineering innovations," according to the statement.

## Hewlett Packard issues a critical open source bug alert

A use-after-free vulnerability that enables remote attackers to execute arbitrary code on targeted systems, leak data, or set up the perfect environment for a denial-of-service (DoS) attack was the subject of a critical alert released by Hewlett Packard Enterprise (HPE) recently in connection with its OneView infrastructure management platform.



The use of Expat XML parser, third-party software, is linked to the bug. With a severity level of 9.8, HPE has assigned the bug the CVE-2022-40674 tracking number. Many other vendors' enterprise-class software have also been harmed by the susceptible code, including NetApp and IBM, both of which have sent customers critical warnings to address the same fault.

There are no publicly available reports indicating that the flaw is being used in the wild or that a proof-of-concept attack has been launched. However, the vendors state there are no mitigations or solutions for the specific Expat fault, despite the fact that IBM and NetApp both offer remedies. Instead, both companies are providing upgrades that protect impacted products.

Eleven of NetApp's enterprise products were affected by the Expat fault, the company informed users recently. And it is still looking into the possibility that host utilities for SAN for Windows may also be impacted.

## Red Hat and Oracle to offer more OS options on Oracle Cloud Infrastructure, beginning with RHEL

Red Hat, Inc. and Oracle have announced a multi-stage partnership to give clients more operating system options to run on Oracle Cloud Infrastructure (OCI). Starting with Red Hat Enterprise Linux (RHEL) running on OCI as a supported operating system, the strategic collaboration enhances the user experience for organisations that depend on both OCI and RHEL to drive digital transformation and the migration of mission-critical applications to the cloud.


Currently, Red Hat and Oracle products are used by 90 per cent of the Fortune 500 companies. RHEL serves as the operating system foundation for many of these businesses, and OCI provides them with high-performance, mission-critical cloud services to power operations that are focused on the future of digital technology. With RHEL operating on OCI, these enterprises can now standardise their cloud operations and have access to a common platform that extends from their data centre to the OCI distributed cloud.

As a result of this strategic partnership, clients can move existing workloads now operating on RHEL to RHEL on OCI with more assurance, and certified configurations of OCI flexible virtual machines can now run Red Hat Enterprise Linux. To improve price-performance and reduce resource waste, OCI flexible virtual machines can scale in steps as little as one CPU. With a more extensive transparent joint support agreement, customers can also contact Red Hat and Oracle support to assist in resolving any difficulties.

With the help of this collaboration, Red Hat and Oracle's clients may now lay the groundwork for computing deployments in the future while still preserving the value of their current IT investments.

## Linux Foundation gets the StarRocks Project

The StarRocks Project, a high-performance analytical database, has been donated to the Linux Foundation by CelerData, a unified analytics platform specifically created for the modern enterprise. As a result, the project will continue to develop and flourish as part of the open source community. This contribution was made following the news in December last year that StarRocks would switch from an Elastic License to an Apache License.

Since its inception in 2020, the StarRocks Project has been a standalone endeavour with publicly accessible source code. More than 500 businesses, including market leaders Lenovo and Airbnb, have successfully launched digital transformation programmes with the aid of StarRocks. It helps developers integrate OLAP, data lakes, and real-time analytics onto a single engine and data pipeline. StarRocks utilises CPU processing power and SIMD (single instruction, multiple data) to improve performance, thanks to its columnar storage engine and fully vectorized operators.

The StarRocks Project has received awards for its product innovation in data analytics. These awards include being named the winner of the BIG Innovation Awards, a finalist in The Cloud Awards for Best Cloud Business Intelligence or Analytics Solution, and the Intellyx's Digital Innovator Award.

"The Linux Foundation is delighted to welcome the StarRocks Project into its family of open source projects," said Mike Woster, Linux Foundation's chief revenue officer. "By providing a neutral home for collaboration, the Linux Foundation is able to bring together talented individuals and organisations from around the world to collaborate on building innovative solutions and technologies for shared benefit."

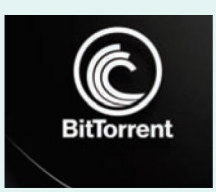
## Version 4.0 of BitTorrent client Transmission released

Version 4.0 of the well-known open source, free, and cross-platform BitTorrent client Transmission has been released. This upgrade brings a host of new features and performance enhancements. The Transmission 4.0 release, which has been in development for more than a year and arrives more than 2.5 years after Transmission 3.0, brings support for BitTorrent v2 and hybrid torrents, support for IPv6 blocklists, and a completely redesigned Web client with full mobile support, including support for dark mode and full-screen.

The ability to set 'default' trackers that can be used to announce all public torrents, and the option to omit potentially identifying information like user-agent and date created when creating new torrents are all new features.

Transmission 4.0 adds customisable anti-brute force settings, the capability to retrieve magnet metadata, support for changing the GTK client's progress bar colour based on the torrent state, an updated 'Details' dialogue that now displays the date a torrent was added, and faster rendering of lengthy file lists.

A new feature called torrent-added-verify-mode allows users to force-verify newly added torrents. Additionally, the Transmission-Qt and Transmission-Web remote control GUIs now employ the RPC API's 'table' mode, which results in smaller payloads and less bandwidth consumption.




The entire code base has also been converted from C to C++, the GTK client has been upgraded to GTK 4 and GTKMM, and the Web client has been completely rewritten in contemporary JavaScript.

## Checkmarx introduces Supply Chain Threat Intelligence for threat identification

Supply Chain Threat Intelligence, which provides detailed threat intelligence on hundreds of thousands of malicious packages, contributor reputation, malicious behaviour, and more, is now available, according to Checkmarx, a leader in developer-centric application security solutions.

Supply Chain Threat Intelligence, based on exclusive research by Checkmarx Labs, provides:

- Identification of harmful packages by attack type, such as dependency confusion, typosquatting, chainjacking, and more.
- Analysing the reputation of contributors by spotting unusual activity in open source packages.
- Information on malicious package behaviour, including static and dynamic analysis of the code to understand how it functions.
- A data lake with over a million packages scanned each month that enables continued research of packages long after they have been purged from package managers.

As an application programming interface (API), Checkmarx Supply Chain Threat Intelligence is supplied in a variety of dashboards and development environments. Users transmit a package name and version, and receive threat intelligence on the package after receiving a special token from Checkmarx.


The API benefits security experts and developers by identifying potential dangers in open source software packages quickly and simply, helping gain insight into the thought process of threat actors, getting information on many packages at once using bulk inquiries, keeping up with cyber threats with real-time updates and notifications on new and developing hazards, and acquiring important context and insights on identified threats to guide security choices.

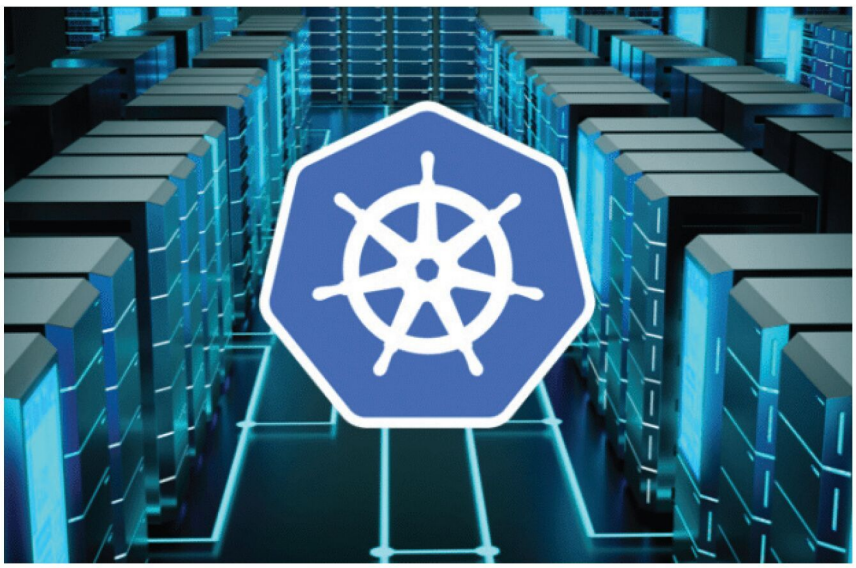
"In 2022, Checkmarx researchers exposed some of the most prolific open source attack groups, including RED-LILI and Lofygang," said Checkmarx CEO Emmanuel Benzaquen. "Given the dramatic proliferation of malicious open source packages from organised attack groups, we're pleased to empower security stakeholders by revealing adversarial motives, tactics, techniques and procedures in a constantly updated intelligence feed."

## Wazuh helps track, archive and index Kubernetes audit logs

Depending on the region and industry in which they operate, corporations must adhere to a number of policies. Some of these regulations, like GDPR,

improve the IT infrastructure's cyber-resilience. Organisations must make sure that the Kubernetes cluster complies with all applicable regulations and security best practices because it is a component of the IT infrastructure. The log retention policy is one of the requirements that may be found in most IT policy documents. How long you should keep logs on file depends on your log retention policy. These logs can be used for incident investigation and active monitoring to find hazards.

To find security dangers and abnormalities, companies must keep an eye on the audit logs. In order to find pertinent information during an incident investigation, they must also index the logs. The Kubernetes audit logs are tracked, archived, and indexed by Wazuh. Wazuh is an integrated XDR and SIEM platform that is open source. It receives more than 10 million downloads annually and is commercial-free.



The Wazuh development team offers a comprehensive manual on using Wazuh to audit Kubernetes. The manual provides instructions for setting up the Wazuh server so that it can receive and handle Kubernetes audit logs.

Kubernetes is regarded as the foundation of application modernisation by enthusiasts. When applications are deployed over multiple servers and containers, their complexity increases. Kubernetes provides an open source API that controls where and how those containers will run in order to manage this complexity.

## Researchers discover more than 700 malicious packages

Another sizable collection of malicious packages, which developers may unknowingly download from the npm and PyPI open source registries, has been uncovered by security experts. Sonatype reported finding 691 malicious npm packages and 49 malicious PyPI components in January this year, both of which contained crypto-miners, remote access Trojans (RATs), and other harmful software.

The same harmful software is included in several packages. A Trojan called *go file* uses Linux systems to mine cryptocurrencies. According to Sonatype, sixteen of these were linked to the same actor, trendava, who has since been taken off the npm registry.

The PyPI malware 'minimums', which is intended to verify the presence of a virtual machine (VM) before execution, was discovered separately.

The security provider also found brand-new Python malware with traits of both a RAT and an information thief. Finally, it discovered 'infinitebrahamanuniverse', a suspicious-looking developer, who uploaded over 33,000 packages that were described as sub-packages of 'no-one-left-behind', or 'nolb'.

## Nix management startup Flox raises US$ 16.5 million in funding

Nix environment management company FloxDev Inc. has raised US$ 16.5 million in new funding and has launched its open source flox platform to build on Nix's distinctive approach to package management and system configuration.

A functional deployment paradigm is used by the cross-platform package manager Nix, which installs software in distinct directories. According to the company, the new flox open source platform offers comfort, teamwork, and control across the system's development life cycle. It accomplishes this by introducing additional features to Nix, including portable, completely isolated development, test, and production environments; a specially curated selection of software packages, and tooling for cycle management.

The platform offers developer, test, and production environments that are created and maintained as code, enabling enterprises to adopt and utilise Nix. Users can simply exchange environments and packages with flox while integrating into current workflows across teams, machines, people, and organisations.

For security and compliance, flox gives full access to the software stack for apps used in production environments, enabling real-time vulnerability management and detection without scanning.

The Series A round, which was headed by New Enterprise Associates, brought the total amount of venture capital funding secured by Flox to US$ 27 million. Aaron Applbaum and Hetz Ventures have both made prior investments.

## The open source ecosystem needs regular funding, say community leaders

According to well-known members of the community, the open source ecosystem will soon require a regular stream of taxpayer financing to fix glaring resource deficiencies. A healthy software foundation will one day fall under the larger 'government purpose', with the public sector actively participating in stewardship, much like how maintaining electrical grids came into its scope a few centuries ago.

According to Amanda Brock, CEO of OpenUK, and Eric Brewer, VP of infrastructure at Google, who talked to IT Pro at State of Open Con 2023, long-standing financing issues in the open source community have widened the gap between heavily used but unmaintained packages that can contain vulnerable code and huge, well-maintained projects like Kubernetes.

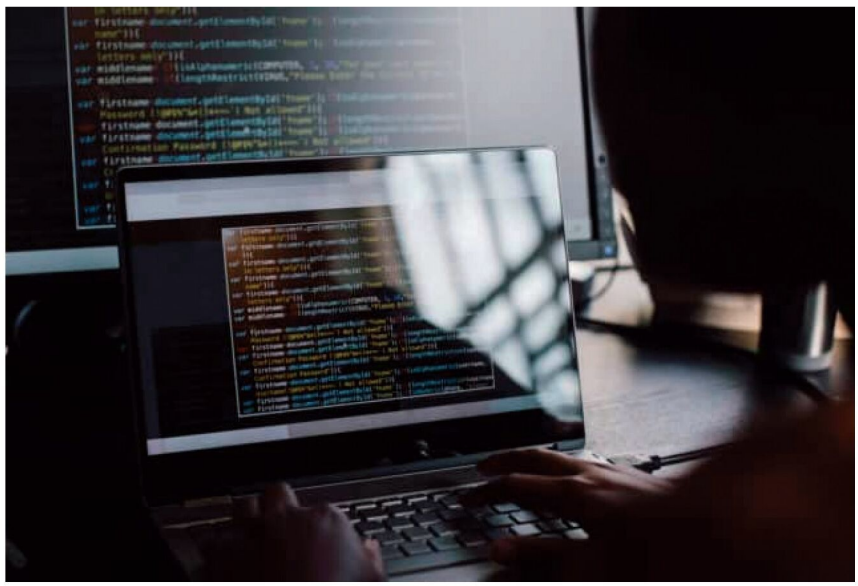
For instance, the Log4Shell exploit in 2021 addressed an undisclosed vulnerability in the widely used Log4j Java logging framework maintained by the Apache Software Foundation (ASF). Many people claimed the project should have been better supported since extra funding and code reviewers might have made a difference.

However, there are divides and arguments within the open source community regarding what the ideal finance and maintenance model might look like in the future, particularly to prevent future security horror stories.

For instance, Rebecca Rumbul, CEO of the Rust Foundation, told attendees at State of Open Con 2023 that governments shouldn't provide the majority or all of the funding for project maintenance. She said that more non-profit foundations, like her own, should be founded and funded in order to act as stewards for initiatives within the ecosystem, even though the public sector and businesses should both play some part.

## VVenC and VVdeC H.266 video encoder and decoder now run on x86 and Arm

Open source H.266/VCC video encoder and decoder VVenC and VVdeC are both optimised for SIMD (single instruction, multiple data) instructions on x86 (SSE42/SIMDe and AVX2) and Arm, while the decoder is compatible with Windows, Linux, macOS, and Android. In 2020, the H.266 video compression standard, also known as VCC (versatile video coding), was approved with the promise of a 50 per cent data reduction over the previous H.265/HEVC standard while maintaining the same visual quality. Since that announcement, there have been no new developments, but the Realtek RTD1319D processor, which was unveiled last September and supports both 4K H.266 and AV1 video decoding, and the advancements made on the VVenC and VVdeC H.266 open source software encoder/decoder, which were discussed at FOSDEM 2023, may be changing that.

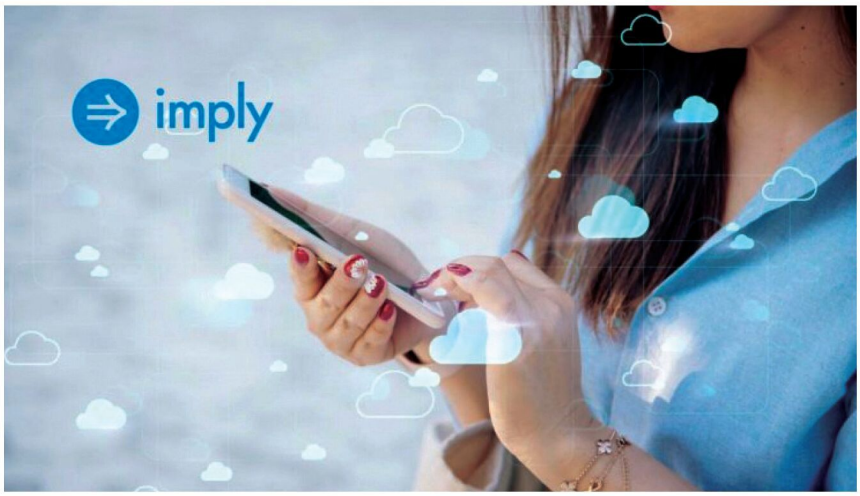


The Fraunhofer HHI group has been working on VVdeC and VVenC since the specifications were finalised in 2022. Both are based on VTM reference software for VCC, are written in C++ with a pure C interface, implement vectorization without the use of an assembler, and are provided under a BSD 3-Clause Clear licence that grants no patent rights. The source code for both is accessible on GitHub. VVdeC is fully Main10 profile compliant, supports more than 30 threads, runs on Windows, Linux (x86, Arm, RISC-V…), macOS (x86 and Arm), and Android. Since the first release, memory usage has been decreased by three times, and the developers are still making incremental advancements.

The VVenC open source H.266 encoder has five settings -- faster, fast, medium, slow, and slower -- each of which offers a balance between quality and encoding speed. It is designed for offline use and VoD (Video-on-Demand) operations. It is now possible to incorporate VVenC and VVdeC into FFmpeg via third-party patches, which enables inclusion into mpv, VLC, and ExoPlayer.

## Imply Polaris wins 'Best Open Source Cloud Solution' award

Imply, the business established by the original developers of Apache Druid, announced that Imply Polaris has won the 'Best Open Source Cloud Solution' honour at The Cloud Awards, a global competition for cloud computing. Polaris offers an easy developer experience for creating real-time analytics apps, as a cloud database solution for Apache Druid.



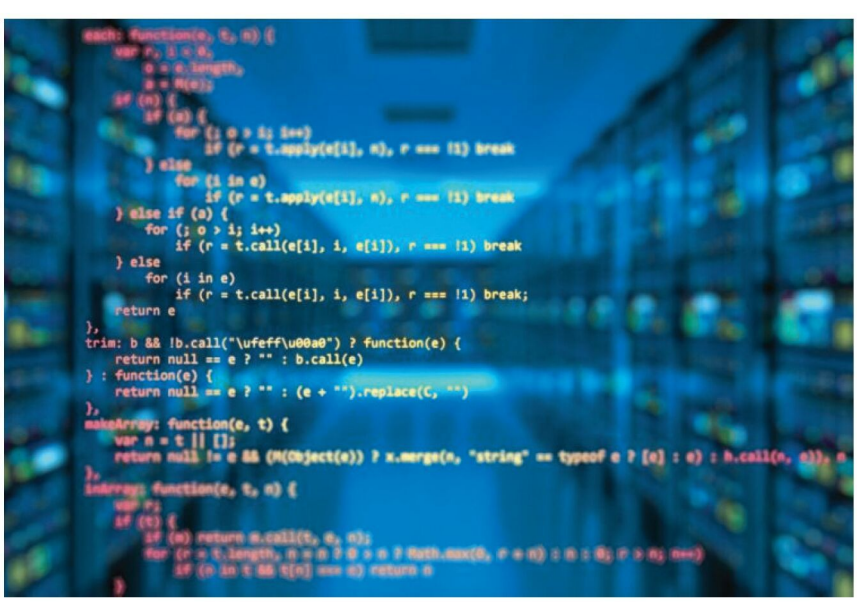

The real-time analytics database Apache Druid, which is used by developers at thousands of companies including Confluent, Netflix, Target, and Salesforce to power real-time analytics applications, has a true database-as-a-service offering in the form of Imply Polaris, which was unveiled in March 2022.

When serving sub-second queries on terabytes to petabytes of streaming, and batch data at hundreds to thousands of queries per second, developers select Apache Druid. They use Imply Polaris as their Druid deployment option because it offers a service that reduces time to market, boosts developer productivity, and lowers Druid operating costs in general.

"We are absolutely thrilled to receive this award," stated FJ Yang, CEO and co-founder of Imply. "It's extra special to us because this award is about open source and cloud — the two things that drive what we do for our customers every day. We believe developers want open source technology, and they want to consume it as cloud services. That's why we built Imply Polaris for Apache Druid, and that's why we're so proud today."

## CISA develops ESXiArgs ransomware recovery tool

The ESXiArgs ransomware attacks, which were discovered for the first time on February 3 this year, take advantage of the high-severity ESXi remote code execution vulnerability CVE-2021-21974 that VMware addressed in February 2021. The flaw is being used by hackers to spread malware that targets virtual machines and encrypts files. There is currently no proof to support the cybercriminals' claims that they have stolen data, despite their threats to disclose it.



Technical information and a proof-of-concept (PoC) exploit for CVE-2021-21974 have been available for almost two years, but up until now there has been no sign of in-the-wild exploitation. Since there is no proof that the ESXiArgs attacks used a zero-day vulnerability, VMware is advising customers to take precautions. There are presently about 2,000 hacked ESXi servers, according to the Censys and Shodan search engines. It's important to note that Censys has found less compromised systems recently, which suggests that affected businesses have been patching up their networks.

An examination of the ESXiArgs attack reveals that after a server has been compromised, the attacker uploads a number of files, including an encryptor, a shell script controlling the attack flow, a public RSA encryption key, and a ransom note, to the */tmp* folder.

BlackBerry researchers conducted an analysis, and found that the shell script is in charge of altering the names of VMX configuration files, terminating VMX processes, locating and encrypting VM-related files, posting the ransom note on the targeted system, and erasing the originals of the encrypted files.

The procedures users must follow to recover their data have been laid forth by security experts Enes Sonmez and Ahmet Aykac. CISA has developed an ESXiArgs ransomware recovery solution that decrypts virtual drives that were not encrypted by the malware using the researchers' tutorial and other publicly accessible information.

For more news, visit *www.opensourceforu.com*

# Will Businesses Benefit from ChatGPT, GPT-3 and DALL-E 2?

Chat GPT, GPT-3 and DALL-E 2 have taken the world by storm. Individuals and businesses must stay updated with these tools, and try and use them well.

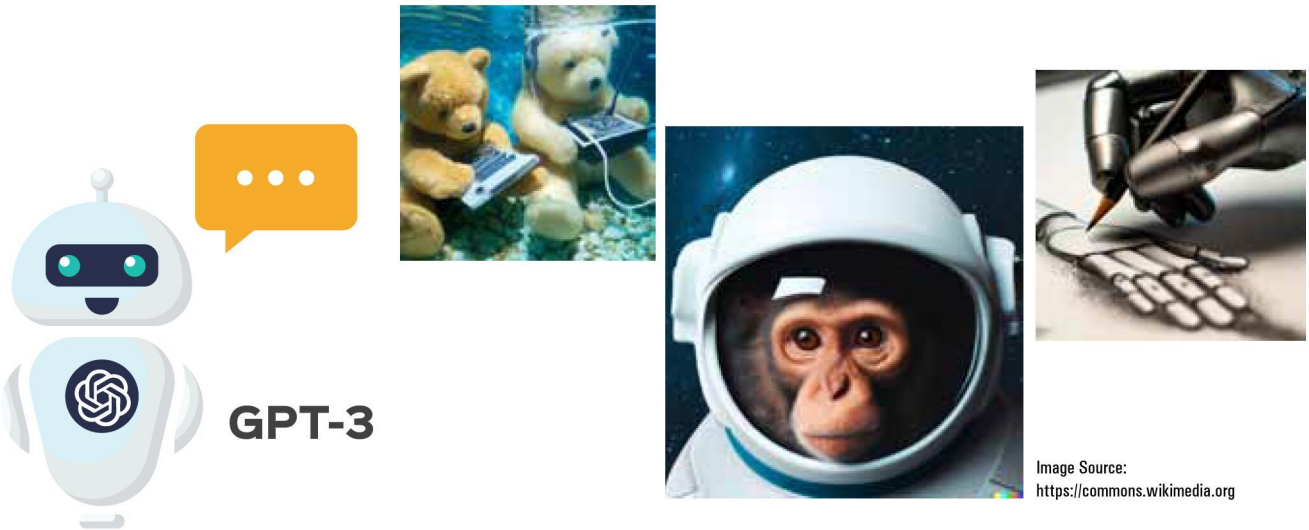



Image Source: https://commons.wikimedia.org

ChatGPT (Generative Pre-Trained Transformer) is an AI chatbot tool that was released by OpenAI in November 2022 and instantly invited people to start conversing with it. Within a few months, over a million users were using it to generate content such as blogs, technical documents, product descriptions, and even to write essays in the style of a specific writer.

## What can ChatGPT do?

Using ChatGPT, you can do anything that involves language and literature. The possibilities are endless. You can create resumes, original jokes, and explain complex topics in a style that anyone will understand. One can write music in almost any genre; explain code, debug it or even create it; and automatically translate blogs into multiple languages or even write blogs on any topic.

## About GPT-3

GPT-3 or Generative Pre-trained Transformer 3 is an artificial intelligence platform created by OpenAI. ChatGPT is a bot that can help create text and simulate real talk with a person, translate speech, etc.

It has 175 billion parameters that make this model one of the most advanced you can find for business use. You can use it to automate customer service, providing a tremendous range of answers to the most popular queries, lowering costs for human operators, and making support faster than ever.

## How can GPT-3 help the business environment?

GPT-3 can be used for creating a website layout, coding some easy features, writing marketing copy, creating blog articles, etc. All this can reduce time to market faster. Process automation is what every business needs nowadays to catch up with the world and meet user requirements.

For example, if you own an online retail store, creating a consultant based on the chatbot can help potential buyers choose sizes, combine clothes to look better, choose proper colours, etc.

Due to the ability of the bot to translate into different languages, it becomes easier to enter new markets and

expand your business. Providing customer support and product description in the language of the country you promote the company in will increase your chances of success.

## OpenAI success story

OpenAI has grown in its technology. High-profile companies like Fidelity Investments, Andreessen Horowitz and Microsoft are showing their interest in building GPT-3 chatbots. Other well-known companies like Tesla, PayPal and Twitter are also backing Open AI technology. Elon Musk, present CEO of Twitter, has invested US$ 10 million in the GPT-3 project.

## Does GPT-3 pose a risk for e-commerce companies?

Any information the OpenAI system receives is then used to further train it. The more people interact with it for various uses, the more it "learns." And that data absorption can include some of a company's sensitive coding and data, which obviously could be very problematic for e-commerce companies.

Recently, an engineer at Amazon asked the chatbot to answer interview questions for a coding position with the company. GPT-3 got all of the technical coding questions correct, which shows just how much GPT-3 has learned about coding in its time online.

## Google and OpenAI

The idea to include chatbots in search engines is new but Google will not be the first to do so. Search engines like NeevaAI and You.com have already beaten Google to this, and both are currently offering that service in beta. Google, the current dominant search engine with billions of users, is likely to adopt the technology once it is more confident about GPT-3's accuracy. Google has invested US$ 400 million in artificial intelligence startup Anthropic to develop a new GPT-3 chatbot.

# DALL-E 2 and the artwork business

DALL-E 2 is an Al system created by OpenAI that automatically creates original art and imagery based on natural language prompts.

This means you type in a description, and DALL-E 2 creates a piece of art or an image based on what you type. You can even tell DALL-E 2 to create the image in a specific style. Ask for an image of 'a Shiba Inu dog wearing a beret and black turtleneck', and DALL-E 2 will create the artwork shown in Figure 1.

Figure 1: Artwork created using AI (Image credit: openai.com)

More than 1.5 million users are now actively creating over 2 million images per day with DALL-E 2. These include artists, creative directors, authors and architects. Over 100,000 users are sharing their creations and feedback in the Open AI community.

DALL-E 2 uses sophisticated deep learning Al called a 'generative model' (powered by neural networks) to not only create images from natural language, but also understand the relationships between objects in the image. It uses a process called 'diffusion', which starts with a pattern of random dots and gradually alters that pattern towards an image when it recognises specific aspects of that image.

However, the flip side of DALL-E 2 is that it could severely disrupt the work and earning power of many conventional designers, artists, photographers, and visual content creators.

ChatGPT, GPT-3 and DALL-E 2 have been presented to the world as an experiment, and users are contributing to their development with their inputs. But companies are using this experimental output in the real world already. It's important to keep in mind that powerful and complex systems like Chat ChatGPT, GPT-3 and DALL-E 2 can be creatively used or misused. What they require is an iterative deployment approach.

These tools can soon become an integral part of our lives, erasing language and art barriers and enabling better understanding of complicated scenarios. END


### References

- Blog: 'What are GPT-3 Chatbots and How to Profit Implementing it For Your Project' by precoders.tech
- Blog: 'What is ChatGPT and how can it be leveraged in marketing ?" by *3dissue.com*
- An article that appeared in *indianexpress.com*
- Blog: 'DALL-E 2 now available without waitlist' by *openai.com*
- Article titled 'DALL-E 2 and the Future of Design' by Mike Kaput



**By: Vinayak R. Adkoli**

The author is a B.E. in industrial production and has served as a lecturer in three different polytechnics for 10 years. He is a freelance writer and cartoonist.

# What is the Metaverse Made Up Of?

There is a lot of hype about the metaverse today. However, it is critical to understand how the metaverse can help incorporate innovative solutions to offer a better customer experience, simplify IT operations, and introduce new age business models. And it's also important to know what technologies together make up the metaverse.


As per a Gartner report, by the year 2026, at least 25 per cent of the world's population will be immersed in activities in the metaverse (shopping, work, virtual class, social media and entertainment). Forrester's 'The State of Metaverse: Look Beyond the Hype to Uncover the Real Opportunities' report claims that more than 50 per cent of all internet users will be early adopters of the metaverse.

## Hyperreal world and metaverse

Hyperreality is a concept that has been in use for more than a decade, where users are made to believe to live in a virtual world. One such classic example is the live shows in Disney World (e.g., Peter Pan) which make you believe you are in that world and you forget the real world. These 3D, 4D and 5D shows use spatial, modelling and immersive technologies.

Hyperreality applications use generative AI algorithms and visual effects like sound, touch sense and real-world objects to create synthetic content which can take users into a virtual world. This kind of synthetic content leads to the development of metaverse platforms through SDK and AI services.

From an architect's perspective, synthetic content in hyperreality, digital twins with *avatars*, and AI generative algorithms for model development have all led to metaverse's success. A unified platform, which combines agility, scalability and reliability through cloud platforms, virtual transaction management through NFTs and blockchain platforms, and immersive technology through human interactive devices can help to build a robust metaverse.

## The technologies that make the metaverse

Let's take a look at the technologies that have come together to make the metaverse.

***Virtual reality (VR) and augmented reality (AR):*** While the metaverse is a platform that is still developing, virtual reality is a technology that is already being used in daily lives. (The metaverse may even replace the internet as we know it today.) Virtual reality can be experienced today in diverse real-time applications like simulators, gaming, Industry 4.0, etc. Metaverse can be experienced via VR and AR. Metaverse technology is far more advanced than virtual reality and is expanding to 3D depiction of the internet or virtual world.

***Blockchain and cryptocurrency:*** 'Crypto metaverses' have a huge social and financial potential, and enable the integration of blockchain infrastructure with the metaverse virtual world. With the fusion of virtual environments using VR, games, social media networking and crypto exchanges, the metaverse could become a 'central element' for the next level of blockchain gaming.

Crypto metaverse is defined as a metaverse that integrates blockchain technology and crypto assets such as 'metaverse tokens'. Examples of crypto metaverses are: Decentraland, The Sandbox, Aliens World and Cryptovoxels. With VR, blockchain, and crypto, the gaming industry can experience a revolution that can be called 'blockchain gaming'. Metaverse crypto assets include digital objects, land, etc, details of which can be recorded on a blockchain and even traded using Bitcoins or Ether on diverse decentralised exchanges (DEXs).

*Key features:*

- *Decentralisation:* Crypto metaverses are decentralised, giving all metaverse games developed on blockchain technology strong value models.
- *Provable provenance:* Crypto tokens called non-fungible tokens (NFTs) will enable transparency and access to asset markets in the gaming industry.
- *Real-world economic value:* Metaverse tokens can be exchanged on DEXs and NFT marketplaces.

***Artificial intelligence:*** Fusion of artificial intelligence (AI) with the metaverse adds value to infrastructure, and gives better information to all the top layers of the metaverse.

Metaverse makes use of AR, VR, AI and blockchain to make advanced replicas of the real world in a virtual world. It applies advanced AI algorithms to perform content analysis, speech processing, computer vision, and more.

This is how the metaverse uses AI.

*Virtual avatars:* With the use of AI, 2D or 3D user images can be scanned and transformed to very realistic *avatars.*

*Digital humans:* Digital humans have human-like capabilities for seeing, listening and understanding the conversation of real humans. They can use digital speech and body language to create human-like conversations. In the metaverse, digital humans can be considered as 3D chatbots that respond just like human beings. They are called 'non-playing characters' whose actions and outcomes are measured by automated scripts or sets of rules.

*Language processing:* With AI, all languages can be transformed to a machine readable format that is analysed, and the output is generated back in the same language. By using advanced ML and DL (deep learning) algorithms, language translation becomes more accurate and fast in response.

*Data learning:* AI helps generate models into which historical data is fed. DL and ML are used to generate new outputs, and as more and more data is fed into the model, the outputs get better.

***3D graphics:*** Metaverse is also dependent on 3D technologies. 3D graphics generate images or objects using 3D graphics software. These can be viewed using specialised hardware called 3D displays, and have diverse applications in the real world.

3D reconstruction is a technology that uses 3D models to visualise an object or environment in a three-dimensional view. The metaverse can only exist within the three-dimensional setting. It's essentially a digital twin of our world.

Meta has rolled out diverse devices like Oculus VR to experience the metaverse.

***The Internet of Things:*** IoT technology provides a strong bridge between the physical and virtual worlds. It helps analyse data from the physical world using smart sensors. IoT and the metaverse can together take smart systems to the next level.

Metaverse can leverage IoT in the following ways.

*'Real' feeling in devices:* Via IoT, the metaverse can make virtual devices look real, and even solve complex problems of hybrid data integration with cloud and digital infrastructures.

*Smart integration:* IoT can help the metaverse collaborate with the real world. It is a strong building block for creating interoperable systems and fusing digital content into the physical environment.

*Data collection:* IoT enables digital twins and virtual simulations in the metaverse. It helps integrate scanned objects with real metadata from the physical environment.

## Universal scene description (USD) for metaverse development

When there are multiple metaverse platforms, there are no uniform technology solutions on how to develop the metaverse platform and the 3D objects in it. Pixar was using universal scene description (USD) for a long time to describe 3D objects, and open sourced it in 2016. Now, for metaverse platform development, many companies like Adobe, NVIDIA, Siemens and Lowe's have started looking into USD as the standard solution for metaverse-based 3D object rendering and development.

Initially, when Pixar used USD, it did so mainly for media related image rendering in the 3D space, in the entertainment sector. We can now expect more real-time use cases across different sectors like medical and healthcare, robotics and retail. USD is more than animation and simulation of

real-world objects — it enhances the immersive experience for end users through its 3D rendering.

When USD grows in the metaverse world as a standard facility, Forrester predicts that growth and standardisation of the four technologies listed below can elevate the growth of the metaverse.

**Extended reality (XR)** is a combination of VR, AR and mixed reality (MR). The continual convergence of these technologies will enhance the immersive user experience.

**Web 3.0** will enable delivery of next-generation web technology solutions. NFTs and secured blockchain based network solutions will help create resilient, high performing and secure platforms for metaverse based cryptocurrency transactions.

**Zero Trust Edge (ZTE)** will provide seamless and uninterrupted network facilities for metaverse solutions. It will combine high performance in network communications with zero or low latency and have embedded security features.

**TuringBot** is a term coined by Forrester for automatic and autonomous code development and testing automation. It can solve problems and develop solutions based on in-built AI-powered software.

## Token economy and metaverse solutions

We have been hearing about NFTs and cryptocurrencies in recent times, but they are misconceptualised as an investment option and are interchangeable. There is also a thin line drawn between Web 3.0, token economy and digital currencies in association with NFTs and cryptocurrencies. It is better to understand each of them to understand how and when to use them.

Token economy refers to a digital transformation process using blockchain technology, where physical assets are digitised and processed in a blockchain platform for online trading. In simple terms, blockchain platform enables token economy models using cryptocurrency tokens. Smart contracts are used to programmatically define how tokens can be managed, exchanged and realised in a blockchain platform.

Figure 1: Technology convergence for metaverse solution

## Industrial metaverse

According to a report from Market Prospects, the industrial metaverse is expected to touch US$ 540 billion in revenue by 2025.

The industrial metaverse prepares current industrial infrastructure for digital transformation. It gives a central platform for companies, offering security, easy accessibility and usability.

Industrial metaverse integrates all processes and applications, bringing everything under a single roof, and making it transparent and manageable.

Companies can fully utilise and leverage the potential of industrial metaverse in the following ways:

1. Brand awareness via factory walks and virtual interactions
2. Virtual presentations of all sorts of products and services
3. Payment via blockchain/NFTs or metaverse's crypto
4. R&D collaborations using digital twins for design, simulation and testing of new products
5. Strong feedback services using customer's voice

The industrial metaverse is still at an infant stage, and it faces challenges because of high equipment costs and technical difficulties.

The idea of an industrial metaverse has attracted a great deal of interest over the last 12 months. It describes, in short, a highly immersive and connected virtual and physical reality enabling never-before-seen levels of intelligent connectivity and AI. END

### References


- *https://mfame.guru/are-we-ready-to-live-in-hyperreality-metaverse/*
- *https://www.abiresearch.com/blogs/2022/12/09/technologies-powering-the-metaverse/*
- *https://cmr.berkeley.edu/2022/12/transforming-your-brand-using-the-metaverse-eight-strategic-elements-to-plan-for/#:~:text=The%20eight%20key%20technologies%20in,that%20are%20relevant%20to%20Metaverse*
- *https://www.forrester.com/blogs/show-me-the-metaverse/*
- *https://new.siemens.com/global/en/company/insights/what-is-the-industrial-metaverse-and-why-should-i-care.html*
- *https://www.technologyreview.com/2022/12/05/1063828/the-industrial-metaverse-a-game-changer-for-operational-technology/*
- *https://siliconangle.com/2022/12/24/industrial-metaverse-will-transform-manufacturing/*



### By: Dr Anand Nayyar and Dr Magesh Kasthuri

**Dr Anand Nayyar** is a PhD in wireless sensor networks and swarms intelligence. He works at Duy Tan University, Vietnam, and loves to explore open source technologies, IoT, cloud computing, deep learning and cyber security.

**Dr Magesh Kasthuri** is a senior distinguished member of the technical staff and principal consultant at Wipro Ltd. This article expresses his views and not that of Wipro.

# Disrupting the Industrial Metaverse Using Open Private 5G and Edge

5G is seemingly the next big thing of the current decade, so it is about time the world adapts to its offerings and moves ahead along with it. Let us try to understand how it will affect the ongoing industrial revolution or Industry 4.0.

Today, manufacturing is the first thought that comes to our minds when we look at an industrial setup that predominantly consists of mining and power grid. In order to digitise the industry entirely you need to ensure that all the Internet of Things (IoT) devices, robots, and drones across the entire country are connected. You need to run real-time applications to enable the strict need for industrial applications, hence, creating a need for reliable connectivity that supports high bandwidth and low latency at the same time.

Let us look at a few reasons for requiring these applications:

*Enhanced mobile broadband:*

- 20/10 Gbps DL/UL
- 4ms user plane latency
- Mobility up to 500km/hr

*Ultra-reliable low latency communication:*

- 1ms user plane latency
- Highly secure/resilient
- 0ms mobile interruption time/ always available

*Massive machine-type communication:*

- 1 million devices per km square
- 10+ years battery life
- 20dB coverage enhancement

## Private 5G and edge cloud may disrupt emerging technologies

A Wi-Fi network is most suitable for the indoors, but it is not scalable when you have to cover it on a 10 square kilometre area. To do that you will have to put a Wi-Fi modem at every 10 to 20 metres, which is impractical and costly at the same time. In this case, a cellular network is the most appropriate option.

We already have working 4G networks. These networks are

predominantly public networks and not private networks with an enterprise. Similarly, cloud is also not physically present within an enterprise.

You cannot be certain of the data centre location. Even if you did know about its location, you will still not be able to know the amount of latency it will take, because it is going through the web network and not through the public network. This is why this latency needs to be changed.

With the introduction of 5G this can be dealt with easily. It allows you to create an alternative Wi-Fi network within the enterprises. It ensures low latency and high throughput as well as privacy.

Along with that, you also need to have a miniature version of the cloud, which we call the 'H' cloud. It is more reliable for things like augmented reality (AR) and virtual reality (VR), video analytics, robotic operations, and autonomous vehicles, as it guarantees low latency. It is also important to note that the cloud is essentially for bookkeeping and data management and not for any computation tasks.

Information technology (IT) applications such as Zoom, system applications and products (SAP) can be run properly on Wi-Fi but operational applications cannot. This is almost a 517 billion dollar market, which will be disrupted eventually.

## Industry 4.0 use cases

Industry 4.0 has many use cases, some of which are mentioned below.

***Logistics:*** Warehouses tend to be extremely huge, hence requiring the need for digitising them as real-time tracking of the devices across the warehouse is extremely important. Drones can be used for tracking purposes while robotic automations can assist with moving the goods from one place to another. AR/VR based applications can be used to assist the training of the employees in real time.

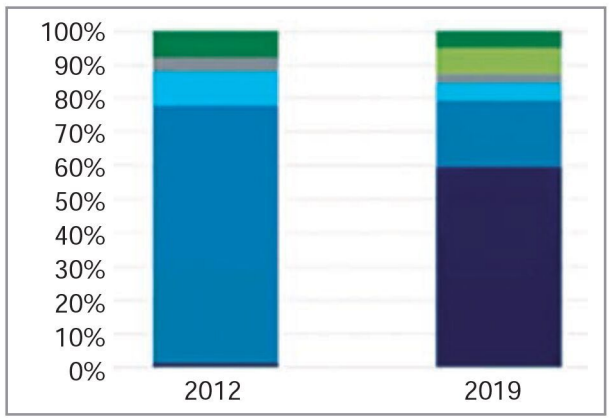



Figure 1: Data centre networking transformation

***Mining:*** Mining in India is of two kinds. First one is called open-pit mining. This kind of mining does not require you to get into a tunnel. You essentially do the mining in the periphery and do the extraction.

The other kind of mining is called closed-pit mining. This method of mining requires you to go underground inside a tunnel. There can be around 10 to 22 tunnels as you go down, and each of them can be one to two kilometres long.

In places like Australia, a lot of automation has been done already. But in India things are still done manually. Miners enter closed pits and use a USB stick to take the data out from the sensors installed in that area. Only then can they connect it to their office systems and servers to use it.

This creates opportunities to digitise everything. By automating the unmanned vehicles this problem of data extraction can be solved so that no human has to enter these mines. This would significantly reduce the risk of any kind of disasters.

You cannot send people inside the mines for causing a blast for a specific purpose. In such cases you can send drones equipped with 4K cameras and take a survey of that area.

***Oil and gas:*** Oil and gas exploration is divided into three predominant parts, namely, upstream - offshore, midstream - pipeline, and downstream - refinery. During upstream - offshore exploration you are basically looking for places where you can find extra oil. Similarly, there is onshore exploration where the exploration is done on land, unlike offshore.

Oil and gas exploration requires a lot of systematic data collection. Heavy analytics is required to find out the right places to conduct these explorations. Private 5G can play an important role to achieve the same. This makes a huge scope to increase productivity and save lives.

***Manufacturing:*** There is a real use case at Fujitsu where they use private 5G network as well as edge computing for autonomous guided vehicles. They use 4K cameras to ensure that these vehicles are moving properly. They collect the data from these 4K cameras as well as sensors, which allows them to know about their real-time movement.

Similarly, in the manufacturing assembly lines they use cameras to ensure the work is being done the right way. They use artificial intelligence (AI) analytics tools to monitor everything. If they find that someone is not doing the work correctly, the AI gives a proper feedback about it. Training is also being done on the go with the help of AI in real time.

## Transformation towards openness and disaggregation

Data centres of companies like Facebook and Google have moved to enable openness and disaggregation, especially in the networking space. Back in 2012, networking within most of the data centres that required switches, routers, and other network appliances were all properties, which

means that they were owned by the original equipment manufacturers (OEMs). They wanted to run an operating system on multiple pieces of hardware, which was not possible at all.

Some of the hyperscalers like Facebook and Google found an opportunity and started using some of the white box hardware. White box hardware can be commercially used right off the shelf. These are servers which are based on open hardware designs that you can easily procure.

After you have taken the white box hardware you can then basically put any open software in there. Now, open software can be taken from various vendors and can still run on the white box made by any of the multiple companies. You can even change the software later, if you need to, and still be able to run it on the hardware. This is also called disaggregation.

The movement from proprietary solutions to open and disaggregated software and hardware in the data centres started in 2012 (see Figure 1).

The white box use was very miniscule at about 2% in 2012. But by 2019 the white box use dominated the market and became around 59%. This was an eye opener for the telecom industry and a lot of enterprises too. Since then, disaggregation and openness has become an effective standard in the data centre market.

This was also the time when our company (Niral Networks) started to envisage what it wanted to do. Interestingly, even big companies like Cisco are moving towards disaggregation, because that is where the market is headed.

## Transformation towards openness and disaggregation in 5G RAN and core

Initially, the radio access network (RAN) was proprietary but with the introduction of open radio access network (O-RAN) it has been divided into three parts now. The first is the radio unit (RU) which acts like antennas. Then there is a distributed unit (DU) and a centralised unit (CU). Now, instead of making use of the hardware you can simply use the cloud to run DU and CU.

The important parts in the 5G are the radio network, as mentioned previously, and the transport network that sends the radio signals to the centralised core. There is a transport network between the 5G network and the centralised cloud using a router which is also getting disaggregated.

## NiralOS network operating system

- Private 5G Core Cloud is a native private 5G core software for mobility, authentication, security, session, and policy management. It contains the 5G network functions like AMF, SMF, AUSF, DM, NRF, and UPF. Niral 5G core also has a compact user plane function (UPF) to provide local breakout within an enterprise when integrated with TSP's centralised 5G core.
- Mobile edge cloud Kubernetes and virtualised edge cloud infrastructure creates a mobile edge cloud within an enterprise with open APIs to host third-party applications like AR/VR, robotics, drones, AI/ML, and video analytics for low latency and privacy.
- Centralised controller provides centralised management, orchestration, zero touch provisioning, and monitoring of multiple private 5G networks and mobile edge cloud at various sites. The controller can be hosted in the public cloud to centrally manage and monitor multiple private 5G networks.

## NiralOS specifications

- Release-16 compliant 5G core for private 5G deployment
- 5G network functions like UPF, AMF. SMF, AUSF, UDM, and NRF
- Kubernetes based cloud-native network function
- DPDK+VPP based user plane (UPF) acceleration - linear scaling per CPU core
- UPF local breakout for easy integration to TSP's centralised 5G core
- Support of N1, N2, N3, N4, and service based interfaces for 5G SBA
- 5G core deployed on COTS hardware of various form factors and integrated with 5G radio
- Kubernetes and virtualised cloud agnostic edge cloud to host third-party applications
- Open APIs for integration of third-party applications to Niral 5G core and edge platform
- Web based dashboard for subscriber provisioning configuration and management

We have done some indigenous end-to-end integration with a radio company in India, so we can provide an end-to-end 5G system. We have recently been awarded for innovation in 5G and look forward to achieving more such feats. END

*This article is based on a tech talk by Abhijit Chaudhary of Niral Networks at Open Source Conference 2022 that was organised by Samsung R&D Institute India, Bengaluru, and IEEE ComSoc Bengaluru Chapter. It was transcribed and curated by Laveesh Kocher, a tech enthusiast at EFY with a knack for open source exploration and research.*

**By: Abhijit Chaudhary**

The author is a founder and CEO of Niral Networks

*The article was originally published in the December 2022 issue of Electronics For You.*

# Asia's #1 Open Source Conference

Total registrations

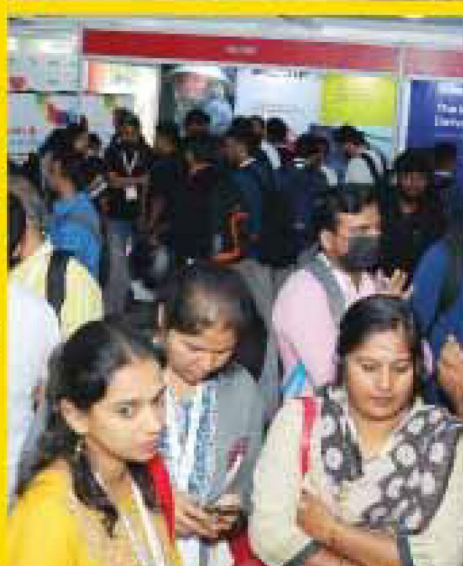
**10,340**

Total attendees

**3732**

Speakers

**90+**



www.OpenSourceIndia.in

An EFYGROUP EVENT

19th Edition

# OPEN SOURCE INDIA

# Thank You

**VISITORS,** for making the event a huge success

**SPEAKERS,** for contributing your valuable thoughts

**PARTNERS,** for your support

Microsoft Azure · Shell · soda foundation · Google Cloud · aws · GitHub · PURESTORAGE · ThriveDX SaaS · harness · DELL Technologies · MOSIP · YML · mt-b · APPLIED SINGULARITY · JAVASCRIPT MEETUP (BANGALORE, INDIA) · OPEN SOURCE BANGALORE Meetup · iotnct · compossible umwelten · APACHE · Apache DevLake · openinventionnetwork · SODA FOUNDATION INDIA MEETUP

For more details, call on +91-98111-55335 or email us at info@opensourceindia.in

# Establishing a 5G Testbed Using Open Source Technology

Open source plays a very crucial part in the world of technology. Every piece of technology today is either fundamentally dependent on it to leverage its growth or to become completely operational. Let us see how it fares with the development of the 5G testbed.

The open source effort to establish a 5G testbed is part of a project that was founded by the Department of Telecommunications of the government of India. It has been running for about four years now. There were nine principal investigators on this project.

The project itself officially ended on 1st January 2022. Now, at the Indian Institute of Science in Bengaluru we are on to the maintenance of the project where we maintain the testbed and encourage its usage and do minor developments.

The following is the overview of different aspects of what this project is all about. There are fundamentally four parts to it. Let us dive in and understand about these attributes.

## 1. System development in sub 6GHz (FR1)

The 6-gigahertz (GHz) development platform is the portion that is actually open source. We have primarily looked at physical layer enhancements and implemented the split six architecture, also called algorithms, and integration with different radio units. We have developed antenna arrays 16 by 4 that are 6499 arrays with an up-down converter and transceiver for the sake of completeness, while testing it alongside extensively in an antiquated chamber.

## 2. Vehicle-to-X (V2X) platform

This is a Long Term Evolution (LTE) based platform. It is a fully functional platform, which can perform end-to-end latency measurements. We have enhanced the MAC schedulers in LTE and reduced the latency of a native implementation of LTE by a factor between 5 and 10. Also, the remote driving capabilities have been demonstrated on campus.

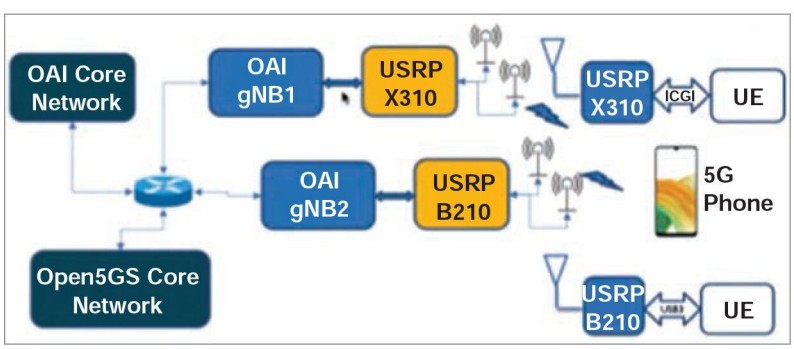



Figure 1: Lab system: connection model

## 3. Visible light communications (VLC) system development

This is a complementary technology to 5G and not a mainstream one. We have built a communication system which is able to deliver about 4.8 gigabytes per second using OFDM. It also incorporates beam steering and has technical knowledge to create its backbone.

## 4. Features of Sub-6 system

We are using a generic radio platform based on the universal serial radio port. This can be operated anywhere up to 6GHz but is primarily focused on the 3.3GHz to 3.8GHz frequency band. The other features are all supported by our testbed, including 30kHz carrier spacing. It has around 100MHz bandwidth and up to 256 QAM. The value proposition of this testbed is that it is a fully open sourced platform, so every part of the 5G radio stack is available to be modified. It has been well tested with USRP X310 and B210 radios as well as with commercial phones like the OnePlus 9 Pro 5G and Oppo A74.

## Lab system: connection model

Essentially, we have USRP radios that are connected to a personal computer (PC) where the gNB1 software stack will be running in a monolithic implementation. Then they are

connected over the network as well.

We have tested it with two different types of cores as well, namely, OpenAirInterface (OAI) 5G core and the Open 5GS core. These cores run on different systems.

The USRPs (refer Figure 1) act as gNB base stations and can be connected to either user's equipment software. It could connect to another USRP and also to another 5G phone.

## Experiments performed so far

We have conducted multiple experiments that include hands-on experiments with the RAN. We have looked at the impact of different parameter settings. We have also looked at different types of real and simulated systems with different types of antennas and how that affects the connectivity and the data rates thus procured.

Core network allows us to analyse various entities and understand the configuration of AMF, SMF, UPF, etc. It also allows us to verify the sequence of events logged at gNodeB and CN. It provides us with a better understanding of SIM card functions.

## TSDSI interaction and work in progress

The vision of TSDSI is to ensure that digital communication standards increasingly drive domestic, economic, and policy activities and enhance India's competitiveness for ICT goods and services in global markets. It aims to do this by creating a leadership position through India's participation and contribution to emerging digital communication standards in global SDOs.

Recently, TSDSI SGI Standard & NaviC (Satellite based Indian Navigation System) have become part of 3GPP. The CPRI Fronthaul Transport standard developed by TSDSI has been taken up by TEC for adoption as a National Standard. Open source begins where standardisation efforts end. They both work hand in hand and can not only co-exist but also augment each other.

**5G Hackathon India**

We contributed to the 5G Hackathon India where our three entries were selected among top 100 out of about 1500 entries:

- Physical Layer Enhancements for 5G in the Indian Context (Selected among TOP 30)
- Advanced MM-wAve Systems for INformatics at Gigabit (AMMAZING) (Selected among TOP 30)
- Robust Tele-Driving over a 5G Network (Selected among the TOP 100)

## TSDSI role in open source

TSDSI has also established an open source task force to study and evaluate open source components required for building an end-to-end system. It explored how open source projects can accelerate the Indian 5G ecosystem. Weekly meetings were conducted for over nine months with broad industry as well as academic participation. The findings were published in August 2021. Some potential impacts are:

- It is a game changer for original equipment manufacturers (OEMs) and service providers.
- Expedites SGI for rural India.
- It reduces the cost of implementation by about 50-60%.
- Comprehensive, well tested open source platform with support.

Building such an open source platform, which can actually reduce the development cost for OEMs, allows project developers to focus on their core competence. We want to keep our approach similar to that of the Red Hat model. There will be basic software available for free and you will have to pay in order to avail more services.

The overall idea is to have three pillars on which this will be built:

- Technical leadership from academia with well-experienced faculty from academia who have experience with several successful projects.
- Dedicated core development team consisting of a team with prior industry experience.
- A strong industry participation through volunteerism to take on specific sub projects and projects of interest.

## IOS - 5G scope

So, in simple terms, there are just two deliverables which we are aiming at. The first one is the DU/CU and the second is the core 5G part of the network. The second part is going to be the RAM intelligent controller (RIC) and the service management orchestration in a cloud based deployment.

In the end, it is important to note that the 5G testbed is open for people to use. In order to use the testbed you can visit *5Gtestbed.in* where all the currently offered services can be found. One just has to register their company and then they can start using the testbed. END

*This article is put together from a tech talk session by Dr Chandra R. Murthy of the Indian Institute of Science at Open Source Conference 2022: Proliferation of Open Source for Emerging Technologies, organised by Samsung R&D Institute India - Bangalore and IEEE ComSoc Bangalore Chapter. It has been transcribed and curated by Laveesh Kocher, a tech enthusiast at EFY with a knack for open source exploration and research.*

**By: Dr Chandra R. Murthy**

The author is a professor in the Department of Electrical Communication Engineering at the Indian Institute of Science, Bengaluru

*The article was originally published in the December 2022 issue of Electronics For You.*

# FOCUS

## Securing Applications

### WordPress: Addressing the Security Challenge

WordPress remains the most popular content management platform today. However, organisations very often forget to update their websites with the latest version, putting them at a security risk.

*Read more on page.....32*

### AI Tools that Enhance Cloud Security


This article explores the importance of AI and ML in the field of open source security tools, and explains how these tools improve the security of cloud environments. It gives some current examples in the field.

*Read more on page.....38*

### Worried About Cyber Security? Look for AI and ML Based Solutions

As organisations rely more and more on the internet, they face the risk of complex and evolved cyber threats. Open source AI and ML based cyber security solutions are the need of the hour to counter these threats.

*Read more on page.....49*

### Using OpenZeppelin for Developing Secured Smart Contracts

The blockchain has numerous applications but is susceptible to security lapses.OpenZeppelin helps integrate a security audit into blockchain-based algorithms called smart contracts.

*Read more on page.....29*

### Static Application Security Testing (SAST) with SonarQube

SAST stands for static application security testing. It focuses on analysing the source code of an application to identify bugs, security vulnerabilities and code smells. The objective of SAST is to identify these issues early in the software development life cycle before they are identified and exploited in the production environment. SonarQube, a popular open source tool, can help with this.

*Read more on page.....54*

### How to Prevent Cookies from Being Hijacked

Every time you log in to a website, you leave a footprint in the form of cookies. These can be used to gain unauthorised access to the information on your system. Let's take a look at how AES 128 can be used to prevent cookie hijacking.

*Read more on page.....61*

# Using OpenZeppelin for Developing Secured Smart Contracts

The blockchain has numerous applications but is susceptible to security lapses. OpenZeppelin helps integrate a security audit into blockchain-based algorithms called smart contracts.


Blockchains are today being used by government as well as corporate agencies because of the security, integrity and privacy they offer. These security features are provided by decentralised applications and smart contracts. Important applications of blockchain include cryptocurrencies, non-fungible tokens (NFTs), financial transactions, logistics management, e-governance, and many others.

Transaction logs associated with blockchain-based applications are secure and have no scope for hacking. Each and every record in the blockchain is linked to dynamic cryptography, enabling all transactions to be encrypted, thus eliminating the risk of sniffing or hacking.

The distributed ledger in a blockchain is a digital asset that has been copied, synced, and shared across several devices and locations to prevent manipulation by third parties. For instance, a bank can offer a better level of security if it uses distributed ledger technology. The records of the transactions will be kept on one million devices if that bank has a million customers. So instead of just one server, the hacker will have to break into one million devices simultaneously. This is the main benefit of utilising decentralised blockchain technology.

If hackers gain access to a bank's server using a centralised application, they can copy all the customer information and data. That is the primary factor driving the decentralisation of web-based applications by government organisations.

Decentralised applications can be used to safeguard government servers that host land registry records, citizens' data (including Aadhaar in India), permanent account numbers (PAN), and so on.

Key application areas of the blockchain are:

- E-governance
- Banking and finance
- Stock market
- Insurance
- Internet of Things (IoT)
- Smart contracts
- Taxation
- Regulatory compliance and audit
- Voting/Polling/Elections
- Logistics management
- Cryptocurrencies and digital assets
- Non-fungible tokens (NFTs)
- Citizen identity management
- Money laundering protection
- Electronic health records
- Energy

## Smart contracts and their implementation patterns

Smart contracts are blockchain-based algorithms that execute when certain criteria are met. They are often used to automate the implementation of an agreement so that all parties may be confident of the conclusion, without the need for an intermediary or any delay. They can automate action once certain criteria are satisfied. When predefined circumstances have been verified, a network of systems or nodes carries out the actions.

These contracts can be used for multiple applications including paying out money to the appropriate receiver, vehicle registration, sending out notices, or booking of tickets. When the transaction is finished, the blockchain is updated. As a result, the transaction cannot be modified, and only parties to whom permission has been granted can view the outcome.

A smart contract can include as many conditions as are required to reassure the participants that the activity will be carried out successfully. Participants must agree on the specific rules that govern those transactions, consider all potential exceptions, and design a framework for resolving disputes while setting the terms. Participants must also decide how transactions and their data are recorded on the blockchain.

A developer can then construct the smart contract. However, more and more businesses are using templates, web interfaces, and other online tools to make it easier to create smart contracts.

## Need for secured smart contracts

Smart contracts enable trustworthy transactions and agreements to be made between dispersed, anonymous participants without the need for a centralised authority, a legal system, or an external enforcement mechanism.

The key advantages of using smart contracts are:

- Speed, efficiency and accuracy
- Trust and transparency
- Security
- Resource optimisation

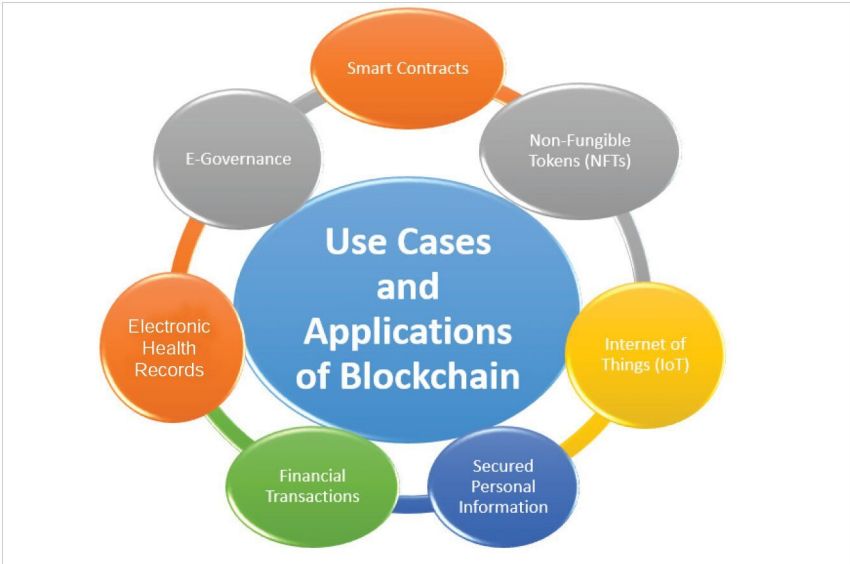



Figure 1: Use cases and application domains of blockchain

Blockchain based smart contracts have the ability to automate various commercial operations, and have numerous applications. However, they also have limitations, such as scalability and security challenges. So decision-makers must balance the benefits against the disadvantages.

Blockchain smart contracts do not need private keys, which are generally required for business blockchain security. Instead, the private keys are controlled by the code that powers smart contracts, allowing data auditing by anonymous users. However, it is possible to further decentralise smart contracts such that they may accept a private key.

Blockchain-based smart contracts are computer programs that run only when certain criteria are met. They are typically used to execute contracts without the need for a third party, letting all parties know exactly what will happen without having to wait for a mediator. They can also automate a process by ensuring that one activity always follows another. And because they run on a decentralised network like blockchain, they are kept in a public database and cannot be changed.

## Using OpenZeppelin for secured smart contracts

One of the basic advantages of using blockchain technology is security. However, smart contracts involve a lot of technical risk and unpredictability. Using OpenZeppelin, the security audit can be integrated into smart contracts. For creating safe smart contracts, OpenZeppelin provides open source OpenZeppelin Contracts written in Solidity. Tokens built on Ethereum and supported by OpenZeppelin Contracts adhere to ERC standards, and may be utilised in a variety of applications. OpenZeppelin Contracts are continuously reviewed and tested in an effort to reduce the cyber risk associated with developing safe decentralised applications on Ethereum or other blockchains.

OpenZeppelin is an open source platform for creating safe smart contracts. Hence, it offers a full range of security solutions and audit services to construct, administer, and examine every facet of software development and maintenance for decentralised applications. OpenZeppelin is maintaining this project with the intent of giving the ecosystem a safe and dependable library of smart contract components.

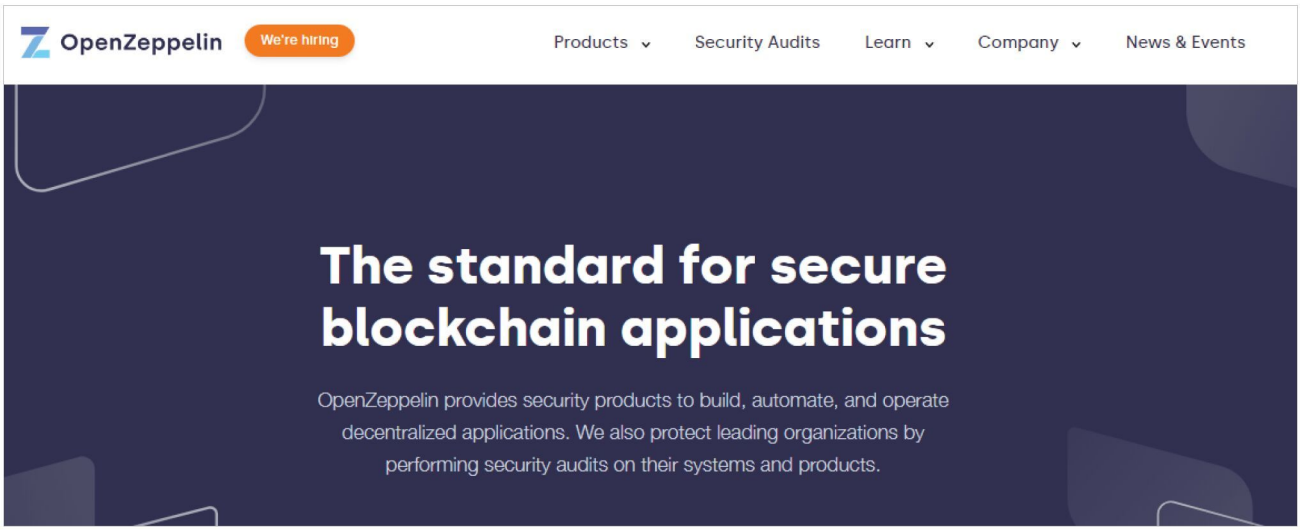



Figure 2: OpenZeppelin for secured smart contracts

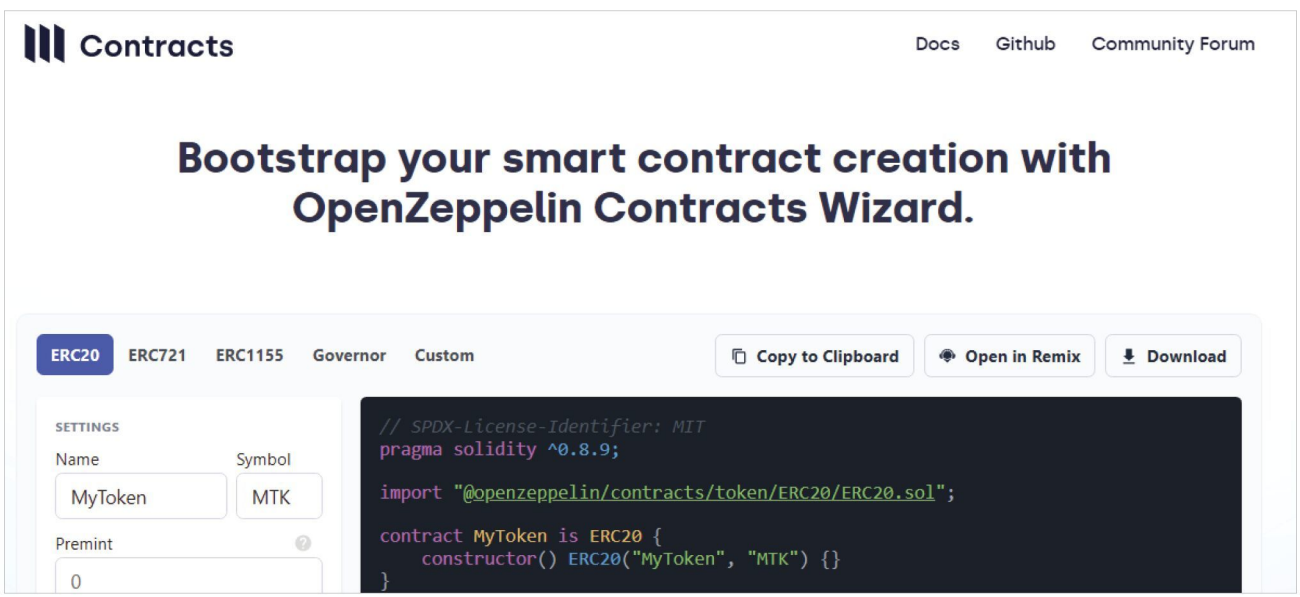



Figure 3: ERC20 smart contract in OpenZeppelin

The MIT License, which restricts the responsibility of individuals who contribute to and maintain OpenZeppelin, and disclaims any guarantees with regard to the project, governs the use of OpenZeppelin Contracts.

As blockchain-based applications are a relatively new development, there are a number of privacy and resource optimisation-related problems that need to be resolved using secured smart contracts. Data replication to massive machines in blockchain technology and other decentralised applications does raise security and integrity concerns. The efficiency of blockchain-based solutions can be improved by creating and implementing sophisticated algorithms with the integration of smart contracts using OpenZeppelin. The blockchain does not totally address security challenges. Despite the primary emphasis on security, abuse has frequently been directed against blockchain-based initiatives like the DeFi (decentralized finance) protocols. This is where OpenZeppelin comes in. The security operations platform aims to deliver a true method for auditing smart contract codes for security risks. END

**By: Dr Gaurav Kumar**

The author is associated with various academic and research institutes for delivering expert lectures and conducting technical workshops on the latest technologies and tools.

# WordPress: Addressing the Security Challenge

WordPress remains the most popular content management platform today. However, organisations very often forget to update their websites with the latest version, putting them at a security risk.

"Content is king! Content is where I expect much of the real money will be made on the internet, just as it was in broadcasting," said Bill Gates nearly 30 years ago. Today, content is at the centre of a business's digital transformation and the customer experience. An organisation must therefore onboard technology platforms that help manage this content, and there is a range of content management systems (CMS) that it can choose from, open source or otherwise.

There has been a lot of evolution in content management systems, be it open source or commercial software. This is one area where open source has not lost its lustre, with frameworks and platforms like WordPress, Joomla and Drupal being the most popular, among many others. Sixty-five per cent of all websites that exist today use these three platforms. That speaks for the robustness these platforms bring to all kinds of use cases across industry domains for small and large organisations. Figure 1 shows the ease of use of the open source CMS platforms that are widely adopted.

Apart from being available for about two decades, these popular platforms have many similarities. They are all based on PHP as a programming language and use MySQL for database storage. All of them use templates and themes backed by robust developer communities that offer rich plugins, modules, and extensions to complement the features in the core platform. All the platforms have feature-rich access

Electronics For You expo
Innovate. Design. Manufacture. Source.

MOVES
MOBILITY & VEHICLE ELECTRONICS SUMMIT

**24TH & 25TH MARCH, 2023** | Auto Cluster Exhibition Center, Pune | India

# Pune Gets An Electronics Expo!


Pune is one of India's fastest growing electronics and automotive regions. The Pune-Nasik-Mumbai triangle represents a market that's not been tapped by any electronics event yet.

Hence, Electronics For You team is taking the initiative to launch **EFY Expo @ Pune**. The expo is powered by **MOVES**—a unique conference targeting mobility, automotive & EV sectors.

So, whether you target automotive / EV clients OR electronics sector at large—you must book a booth at this event to encash this golden opportunity.

**For more information on sponsoring and exhibiting**

**Call:** Ms Mameeta (+91-95998-14784)
**Email:** growmybiz@efy.in

**www.EFYEXPO.com**

controls that offer flexibility and protection, supporting most use cases with different user permissions and capabilities. And they have easy-to-use flexible user interfaces for quick extensibility and customisation.

WordPress, being the most adored platform, has the most plugins, and the largest developer community and market share. However, the popularity and openness of this platform, coupled with its simplicity, attracts hackers to exploit the weakness or poor security of the websites that use it. Although security is a big challenge and concern, staying ahead of the curve by understanding known vulnerabilities and addressing them with appropriate processes does help to keep hackers at bay.

It is critical to install the most up-to-date software version for the core platform along with all the plugins, extensions, and modules. The same applies to WordPress. Over 50 per cent of all sites hosted on WordPress that are hacked are on 'outdated' versions. It proves the point that just adopting a technology and framework is not enough; a process must be set up to ensure the platform is always up-to-date.

A common misconception is that developer self-service refers to a certain life cycle stage of a service or resource, namely, its start. People think they should focus on making it easy to clone templates or spin up a database without having to deal with Terraform. This is a part of the self-service. But I would argue it is a small part. Ask yourself this: How often do you spin up a new service or database? Not very often in the grand scheme of things. The real return on investment of self-service lies in the remaining parts of the service, app, or resource life cycle. This includes making it simple for a new developer to understand what belongs where, enabling clear and separated progression between environments, updating *env* variables effectively, updating resource configurations easily, ensuring security without taking freedom away; and making it super easy to debug deployments and surface error messages as well as consolidate logs. All this gets better through self-service!

| Ease of .. | WordPress | Joomla | Drupal |
|---|---|---|---|
| Use (Market Share) | 65% | 15% | 10% |
| Add text content | ✓ | ✓ | ✓ |
| Add media content | ✓ | | |
| Maintenance | ✓ | ✓ | ✓ |
| Visual templates | ✓ | ✓ | |
| Security | | ✓ | |
| Software Updates | Frequent | Frequent | Less Frequent |
| Plugins/Extensions | 55K+ | 10K+ | 40K+ |

Figure 1: A comparison of the ease of use of popular open source CMS platforms

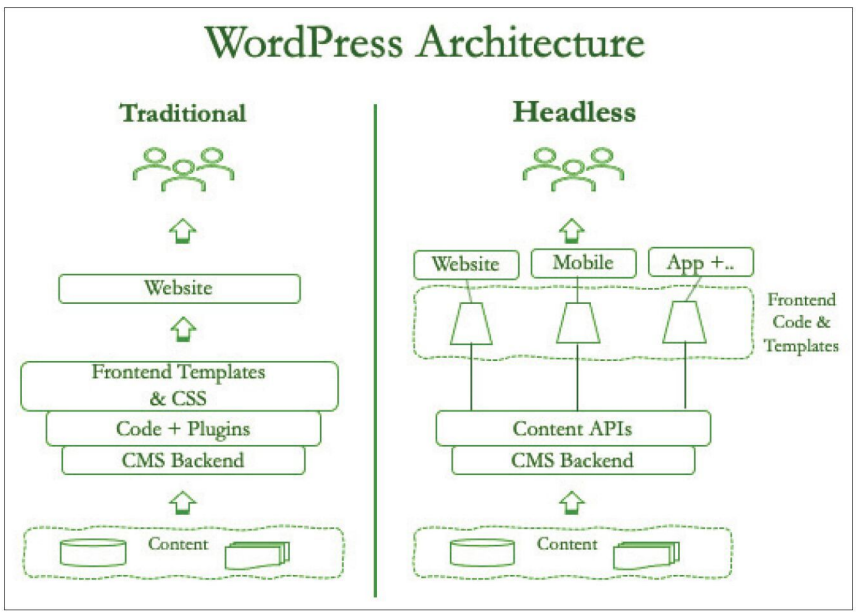



Figure 2: WordPress architecture for traditional and headless implementations

The security of any application is as good as its weakest access point. It's important to know the architecture of the application, especially the deployment model, to ensure the infrastructure, network, and application access points are protected. WordPress architecture has two primary models — the

traditional monolith, where the backend content is deeply integrated with the frontend of the website. Since WordPress is packaged this way, this is the most widely adopted model. However, many organisations, while embracing digital transformation with web, mobile, and other mediums of interfaces, deploy WordPress as a headless architecture. Here, the frontend, middle, and backend are all decoupled to provide greater flexibility of leveraging the same content across different CMSs. This architecture focuses on a seamless and effortless customer experience across multiple channels.

Simple best practices like a secure website (HTTPS), multi-factor authentication, frequent change of admin user name and password, along with proper screening and updation of the plugins in use will ensure hackers and users with malicious intent are kept at bay.

CMS software evolves rapidly. Typically, new versions are distributed more often than most other software. These updates include responses to security and vulnerability threats. Adoption of dynamic configuration management (DCM) frameworks allows developers to separate environmental configurations from that of the platform and application. This allows for rapid deployment of new versions of software with minimal impact on the applications. END

## References


- WordPress.com, *https://wordpress.com/*
- WordPress Community, *https://make.wordpress.org/community/*
- WordPress Security – 19 steps to lockdown your site, *https://kinsta.com/blog/wordpress-security/*
- Wordfence – WordPress Security, *https://www.wordfence.com/blog/category/wordpress-security/*
- MainWP Security Roundtable: Preventing compromises for your WordPress website, *https://mainwp.com/wordpress-security-roundtable-preventing-compromises/*



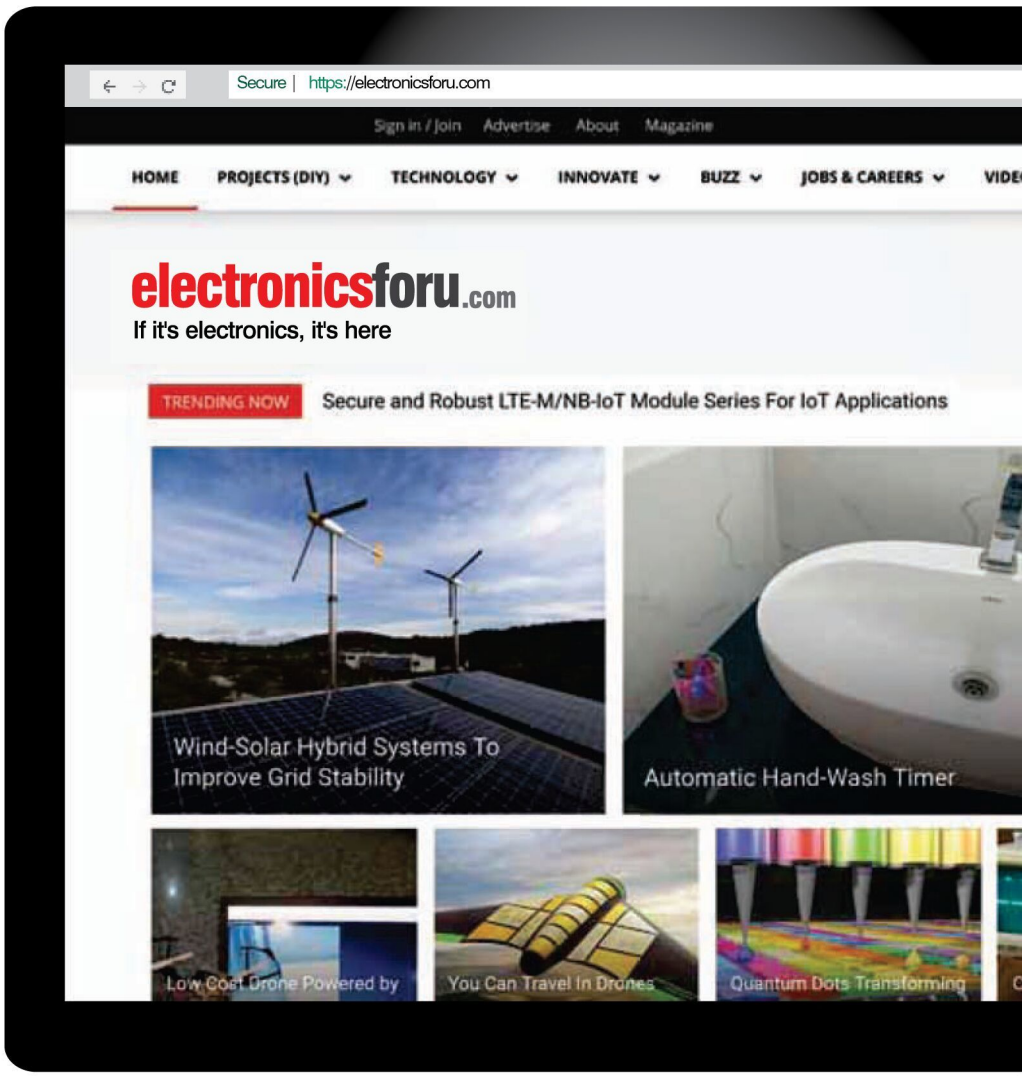
**By: Bala Kalavala**

The author is a technical architect, evangelist, thought leader, and sought-after keynote speaker. He currently works as a distinguished member of the technical staff and head of the Enterprise Architecture practice as chief architect in a global technology consulting firm.


**OSFY Magazine Attractions During 2023-24**

| Month | Theme |
|---|---|
| March 2023 | Security, Network Management and Monitoring |
| April 2023 | Open Source Programming (Languages and tools) |
| May 2023 | Cloud Special: Everything from management to implementation |
| June 2023 | AI, Deep learning and Machine Learning |
| July 2023 | Database management and Optimisation |
| August 2023 | Mobile/Web App Development, Optimisation and Security |
| September 2023 | DevOps Special |
| October 2023 | Blockchain and Open Source |
| November 2023 | Open Source and IoT and Edge |
| December 2023 | All About Data Management |
| January 2024 | Containers and Managing Containers |
| February 2024 | Open Source on Windows and Best in the world of Open Source (Tools and Services) |

# Your favourite website has




# electronicsforu.com

## THANKS TO YOU—OUR ONLINE NETWORK IS

### FACTS & FIGURES

- 4 websites (two more coming soon)
- Five major Facebook communities
- Seven major LinkedIN groups & pages
- Million-plus active users (monthly)
- Million-plus reach through Facebook
- Fifty-thousand-plus industry connections through LinkedIN

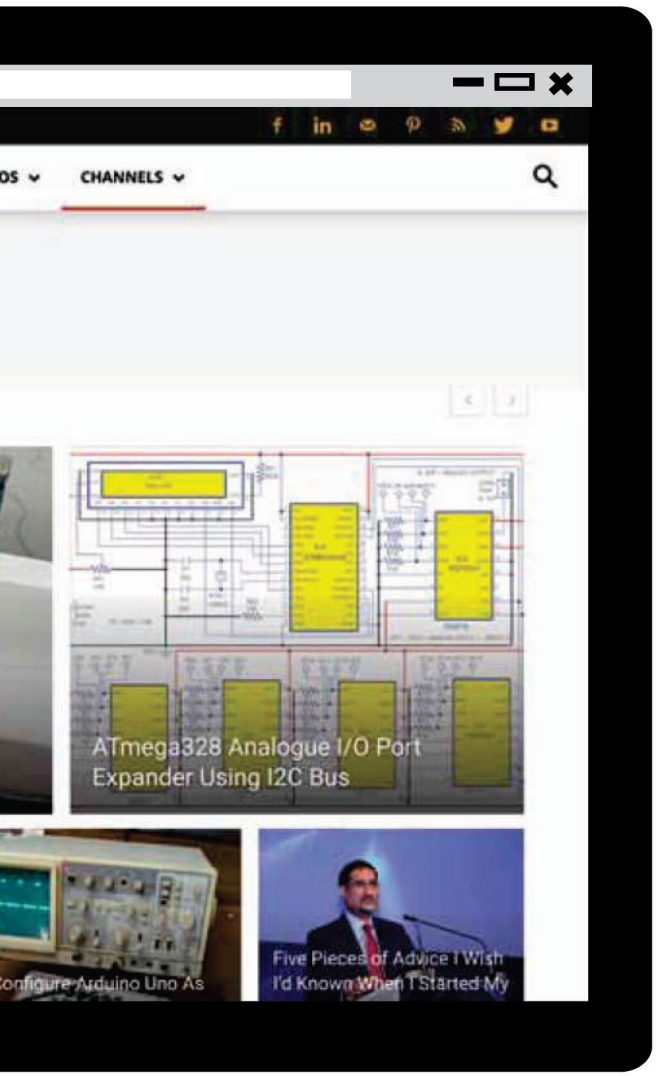
### READERS

- You can access all content for FREE
- You can subscribe to newsletters for FREE --on most websites
- Register on our websites to get free invites to technical webinars and seminars

### EXPERTS

- Experts who want to share their knowledge through articles, DIY Projects, etc are welcome
- We also welcome experts who want to share their knowledge through webinars or seminars
- You can contact us at editop@efy.in

# fast growing peers now...




**Amazing DIY Projects. Latest Tech trends.**
The hang-out for electronics enthusiasts.

**IOT.electronicsforu.com**
The Latest in IOT.
A platform for enablers, creators and providers of IOT solutions.

**DIRECTORY.electronicsforu.com**
India. Electronics. Directory.
Enabling commerce between buyers & sellers of electronics in India.

**BUSINESS.electronicsforu.com**
Business. Electronics. India.
Everything you want to know about India's electronics industry.

## AMONGST THE WORLD'S TOP 5 AND GROWING!

**INDUSTRY**

- You can advertise for as little as US$ 100 per month
- Special combo offers for advertisers in our print publications
- We've now enabled flexible CPM-based advertising
- You can advertise on the platform of your choice (based on your target audience)
- We invite press releases at efy-edit-team@efy.in
- Press releases are published free of cost, subject to discretion of the editorial team

**RESPONSE GUARANTEED SOLUTIONS**

We now also act as marketing partners for our clients and drive entire marketing for them, where we charge them on basis of results and not efforts!

CONTACT US: Shrikant Rao • growmybiz@efy.in • +91-98111 55335

# AI Tools that Enhance Cloud Security

This article explores the importance of AI and ML in the field of open source security tools, and explains how these tools improve the security of cloud environments. It gives some current examples in the field.


The cloud has revolutionised the way organisations store, manage, and access data. However, with the increasing adoption of cloud technology comes a growing need for effective security measures to protect sensitive information from potential threats. In recent years, artificial intelligence (AI) and machine learning (ML) have become key components of cloud security strategies, and their integration into open source security tools has been a game changer.

## What problems in cloud security can AI/ML tools solve?

Open source AI tools can enhance cloud security by detecting potential threats, monitoring network activity, and performing automated responses to prevent breaches. There are four basic problems that can be solved using open source AI tools in the field of cloud security.

**Anomaly detection:** Anomaly detection is a technique used by AI algorithms to identify patterns in data that deviate from normal behaviour. In the context of cloud security, this can be applied to detect unusual activity on a network, such as a sudden spike in network traffic, or an attempt to access sensitive data. By detecting these anomalies, AI algorithms can alert security teams to potential threats, allowing them to take preventive action before a breach occurs.

**Log analysis:** Log analysis is the process of examining log files generated by systems, applications, and devices to identify patterns and trends. With the sheer amount of data generated by

cloud systems, traditional log analysis techniques are often inadequate. This is where AI-powered log analysis tools can be extremely useful. These tools can quickly analyse large volumes of log data and flag potential security issues, such as unauthorised access attempts, suspicious network activity, or signs of a malware attack.

**Automated threat responses:** Automated threat response refers to the use of AI algorithms to respond to security threats automatically. This can be particularly useful in cloud environments where the scale and complexity of systems can make manual response time-consuming and error-prone. AI algorithms can be programmed to take actions such as blocking malicious IP addresses, shutting down compromised instances, or triggering incident response protocols. By automating the response to threats, organisations can minimise the risk of breaches and the impact of attacks.

**Vulnerability scanning:** Vulnerability scanning is the process of identifying and analysing the security vulnerabilities in a system. In the context of cloud security, AI algorithms can be used to automate the vulnerability scanning process, making it much faster and more efficient. AI algorithms can scan cloud systems in real-time, identify potential vulnerabilities, and prioritise remediation efforts based on the severity of the risk posed. This helps organisations stay ahead of potential security threats, and ensure the ongoing security of their cloud environments.

## Use cases of open source AI tools for cloud security

### Suricata for anomaly detection

Suricata is a powerful open source tool for network security monitoring (NSM) that is being used to detect potential threats in cloud environments. It uses signature-based detection and behavioural analysis to identify anomalies in network traffic that may indicate a security threat. With its ability to generate alerts for these anomalies, Suricata is a valuable tool for cloud security.

Anomaly detection is an important aspect of cloud security as it allows organisations to identify unusual behaviour in network traffic that may indicate a security threat. Suricata uses signature-based detection to match network traffic against known threats. For example, if a user visits a website that is known to be hosting malware, Suricata will generate an alert indicating that the user may be at risk.

Suricata also uses behavioural analysis to identify anomalies in network traffic. For example, if a user accesses a cloud application from an unusual location, Suricata will generate an alert indicating that the user may be at risk. This helps organisations to identify potential security threats in real-time, before they can cause significant harm.

One of the benefits of using Suricata for anomaly detection is that it is highly configurable. This allows organisations to customise the tool to meet their specific security requirements. For example, Suricata can be configured to ignore traffic from trusted sources, such as internal IP addresses, to minimise false positive alerts.

The following code snippet is an example of how Suricata can be used to detect anomalies in network traffic.

```
# Rule to detect traffic from a known
malicious IP address
alert tcp any any -> any any
(msg:"Potential malware detected"; \
flow:established,to_server; \
content:"10.0.0.1"; \
reference:url,www.malware.com; \
classtype:trojan-activity; \
sid:100000; \
rev:1;)
```

In this example, Suricata will generate an alert if it detects traffic from the IP address '10.0.0.1'. This alert will indicate that the traffic may be associated with a known malicious website (*www.malware.com*). This type of rule can be used to detect potential threats in real-time, and help organisations to respond to security incidents more effectively.

Suricata also provides integration with other security tools, such as intrusion detection systems (IDS) and security information and event management (SIEM) platforms. This integration allows organisations to receive alerts from Suricata in real-time, and take appropriate action to protect their cloud environment.

### ELK Stack for log analysis

The ELK Stack, consisting of Elasticsearch, Logstash, and Kibana, is a popular open source solution for log analysis in cloud security. The ELK Stack allows organisations to centralise, analyse, and visualise log data from multiple sources, making it a valuable tool for identifying potential security threats in cloud environments.

One of the key benefits of using the ELK Stack for log analysis is that it provides a centralised repository for log data. Logstash is used to collect log data from various sources and send it to Elasticsearch, where it is stored and indexed. This centralised repository allows organisations to search and analyse log data from multiple sources in one place, making it easier to identify potential security threats.

The following is a simple code snippet that shows how Logstash can be used to collect log data and send it to Elasticsearch.

```
input {
  file {
    path => "/var/log/system.log"
```

```
    start_position => "beginning"
  }
}

output {
  elasticsearch {
    hosts => ["localhost:9200"]
  }
}
```

In this example, Logstash is configured to collect log data from the file '/var/log/system.log' and send it to Elasticsearch. This allows organisations to centralise log data from multiple sources, making it easier to identify potential security threats.

Kibana is used to visualise log data stored in Elasticsearch. With its powerful visualisations and dashboards, Kibana allows organisations to quickly identify patterns and trends in log data, making it easier to identify potential security threats. For example, if Kibana detects an unusual increase in logins from a particular location, it may indicate a security threat and organisations can take appropriate action to protect their cloud environment.

The following is a simple code snippet that shows how Kibana can be used to visualise log data.

```
{
  "aggs": {
    "events_per_hour": {
      "date_histogram": {
        "field": "timestamp",
        "interval": "hour"
      }
    }
  }
}
```

In this example, Kibana is configured to display a histogram of log data based on the 'timestamp' field, grouped by hour. This allows organisations to quickly identify patterns and trends in log data, making it easier to identify potential security threats.

The ELK Stack is highly configurable, allowing organisations to tailor the solution to meet their specific security requirements. For example, organisations can configure Logstash to collect log data from specific sources, and Kibana to display specific visualisations and dashboards.

## OSSEC for automated threat response

OSSEC (Open Source Security) is an open source intrusion detection system that is widely used for automated threat response in cloud security. It provides real-time monitoring and analysis of log data from various sources, making it a valuable tool for identifying potential security threats in cloud environments.

One of the key benefits of using OSSEC for automated threat response is its ability to detect and respond to security threats in real-time. OSSEC uses a set of pre-defined rules to identify potential security threats in log data, and can be configured to take a specific action in response to a threat. For example, if OSSEC detects a successful login from a location with a high rate of malicious activity, it can be configured to block the login and notify the security team.

The following is a simple code snippet that shows how OSSEC can be used to detect and respond to a security threat.

```
<ossec_config>
  <rules>
    <rule id="100000" level="10">
      <if_sid>5501</if_sid>
      <srcip>10.0.0.1</srcip>
      <description>SSH login from a
known malicious IP</description>
      <group>ssh,</group>
      <action type="blocking">
        <remote_command>block 10.0.0.1</
remote_command>
      </action>
    </rule>
  </rules>
</ossec_config>
```

In this example, OSSEC is configured to respond to a security threat (defined by rule ID 100000) by blocking access from a known malicious IP (10.0.0.1) and executing the 'block 10.0.0.1' command. This allows organisations to automate the response to potential security threats, making it easier to protect their cloud environments.

OSSEC is highly customisable, allowing organisations to tailor the solution to meet their specific security requirements. For example, organisations can create custom rules to detect and respond to specific security threats, or configure OSSEC to send notifications to specific individuals or groups.

In addition to its real-time monitoring and analysis capabilities, OSSEC also provides a centralised repository for log data. This repository allows organisations to search and analyse log data from multiple sources in one place, making it easier to identify potential security threats.

OSSEC is designed to be easy to install and use, making it a popular choice for organisations of all sizes. The OSSEC community provides a wealth of resources, including documentation, tutorials, and forums, to help organisations get up and running quickly and easily.

## OpenVas for vulnerability scanning

OpenVAS (Open Vulnerability Assessment System) is a widely used open source solution for vulnerability scanning in cloud security. It provides organisations with the ability to

quickly and easily identify potential security vulnerabilities in their cloud environments, helping them to proactively manage risk and maintain a high level of security.

One of the key benefits of using OpenVAS for vulnerability scanning is its ability to scan a wide range of systems and services, including web applications, databases, and network devices. This makes it an ideal solution for organisations with complex cloud environments that need to ensure the security of multiple systems and services.

The following is a simple code snippet that shows how OpenVAS can be used to scan a web application for vulnerabilities.

```
openvas-scanner --target=<target-url>
--profile=Full-and-fast
```

In this example, the OpenVAS scanner is being used to scan a web application at the specified target URL using the 'Full-and-fast' profile. This profile provides a comprehensive scan that covers a wide range of vulnerabilities, including those related to web applications, databases, and network devices.

OpenVAS provides a wealth of information about potential security vulnerabilities, including the severity of the vulnerability, the type of vulnerability, and the potential impact of the vulnerability. This information can be used to prioritise remediation efforts and help organisations to effectively manage risk.

In addition to its vulnerability scanning capabilities, OpenVAS also provides a centralised repository for vulnerability information, making it easier for organisations to track and manage security risks over time. This centralised repository allows organisations to search and analyse vulnerability information from multiple scans in one place, making it easier to identify trends and patterns in vulnerability data.

OpenVAS is designed to be easy to use, making it an ideal solution for organisations of all sizes. The OpenVAS community provides a wealth of resources, including documentation, tutorials, and forums, to help organisations get up and running quickly and easily.

## The benefits of developing open source AI tools for cloud security

Let us discuss some of the benefits of developing and deploying open source AI/ML tools in the field of cloud security.

1. Open source AI tools are typically free to use, reducing the cost barriers associated with proprietary AI solutions.
2. The open source community can contribute to the development and improvement of AI tools, resulting in faster innovation and improved capabilities.
3. Open source AI tools can be modified and customised to meet specific security needs, providing a more tailored solution for organisations.
4. The open source community can collaborate and share best practices and solutions, improving the overall security posture of the cloud.
5. Open source AI tools provide access to the source code, allowing organisations to better understand how the tool works and how it can be improved.
6. The open source community can identify and fix security vulnerabilities in AI tools, improving the overall security of the cloud.
7. Open source AI tools can integrate with other open source security solutions, improving the overall security ecosystem.
8. The open source community can test and validate AI tools, improving reliability and reducing the risk of downtime.
9. Open source AI tools benefit from a large and active community of users, developers, and contributors, providing a wealth of knowledge and resources.
10. Open source AI tools can be adopted by a wider range of organisations, including those with limited budgets, making cloud security more accessible and inclusive.

The use of AI and open source tools in cloud security provides numerous benefits, including improved threat detection, increased efficiency, enhanced accuracy, real-time response, and better compliance management. Open source AI tools offer an affordable and flexible solution, providing organisations with the ability to customise and improve their security posture. With the active collaboration of the open source community, AI tools for cloud security are continually evolving, providing organisations with the latest security capabilities and threat intelligence. The benefits of using AI and open source tools in cloud security make them a valuable addition to any organisation's security toolkit. END

By: Mir H.S. Quadri

The author is a research analyst with a specialisation in artificial intelligence and machine learning. He is the founder of Arkinfo, which focuses on the research and development of tech products using new age technologies. Being a FOSS enthusiast, he has contributed to several open source projects.

# Dynamic Application Security Testing Using Acunetix and GuardRails

Dynamic application security testing (DAST) focuses on finding security vulnerabilities in a web application while it is running. This article looks at Acunetix and GuardRails, which are two popular DAST tools.


Security controls are integrated into the DevOps process through a method called DevSecOps. This includes integrating security early in the software development life cycle (SDLC). Additionally, DevSecOps enhances cooperation between the development and operations teams by including security teams into the software delivery process. Security becomes a shared responsibility under DevSecOps, which calls for changes in technologies, processes, and culture across the SDLC key functional teams. The DevOps continuous integration and continuous delivery (CI/CD) process must incorporate security, and everyone participating in the SDLC has a part to play.

Testing, triage, and risk mitigation should be included early in the CI/CD workflow to avoid the time-consuming and frequently expensive consequences of making a patch after the event. This idea is a component of 'shifting left', which brings security testing closer to programmers so that they may address security flaws in their code almost immediately rather than 'bolting on security' at the end of the software development life cycle (SDLC). DevSecOps integrates real-time continuous feedback loops and insights throughout the whole SDLC, from planning and design through coding, building, testing, and release.

## Dynamic application security testing (DAST)

Dynamic application security testing (DAST) is used to investigate a web application and find vulnerabilities using simulated attacks. This type of technique evaluates the software from the 'outside in' by attacking an application just as a malicious user might. A DAST scanner looks for results that do not match the intended result set after the execution of these attacks in order to identify security issues. DAST has the advantage of identifying potential security holes without taking the application into account. And it does not require access to the source code. The disadvantages are that it cannot determine the exact position of a code vulnerability and that security knowledge is needed to interpret reports.

Software applications are evaluated using DAST. This testing simulates the actions of a malicious party trying to access an application remotely. Software applications are continuously scanned by DAST for security holes using market-leading vulnerability sources like the Open Web Application Security Project (OWASP Top 10), Common Weakness Enumeration (CWE) and SysAdmin, Audit, Network and Security (SANS). OWASP is a standard document for developers and web security; it includes injection (SQL injections, command injections, CRLF injections, and LDAP injections), broken authentication, sensitive data exposure, XML external entities, broken access control, security misconfiguration, cross-site scripting, insecure deserialization, using components with known vulnerabilities, and insufficient logging and monitoring.

The main difference between DAST and SAST (static application security testing) is how each approaches security testing. SAST scans the application code while it is at rest to look for defective code that provides a security danger, whereas DAST analyses the operational program without having access to its source code.

DAST is a type of closed-box testing that simulates the viewpoint of an external attacker. It is assumed that the tester is unaware of the inner workings of the program. It can find security flaws that manifest themselves only during software runtime.

## Acunetix

One of the DAST tools is Acunetix. Although there are numerous tools available today, Acunetix offers a free trial as well as membership services for premium features like speed and accuracy. It is a superior tool that is quick and accurate, and it also has a user-friendly interface that makes it simple to operate and utilise for our website. Acunetix is a program that analyses web applications for exploitable defects like SQL injection and cross-site scripting. It is used to evaluate the security of online applications. Acunetix typically scans any website or online application that may be accessed using a web browser and uses the HTTP/HTTPS protocol.

Acunetix offers a powerful and distinctive approach to evaluating pre-built and bespoke online applications, including those that make use of JavaScript, AJAX, and Web 2.0 technologies. A sophisticated crawler on Acunetix can locate practically any file. This is crucial, because something cannot be verified without being found. Finding a bug is crucial for making a perfect application.

## Configuring and working of Acunetix

Begin by downloading Acunetix from the internet. The link is *https://acunetix-web-vulnerability-scanner.software.informer.com/download/#downloading.* It has a trial version, but is a subscription based application.


Figure 1: Login page of Acunetix

Next, install the application. You need to create a profile with user name and password. We can see the login page in Figure 1.

The application gets opened in '13443' port by default. In case a process is running on that port, you can change it while starting the application.

After you log in, you will be taken to the dashboard, where you can see your overall test results. Figure 2 shows an example of a dashboard.

For running a new scan, go to *Targets* and then click on 'Add Target'. A pop-up appears as shown in Figure 3.

Enter the hosted address of the application and description. Then click on 'Add Target'.

After adding *Target,* you will be taken to a page where you can select scan speed, and business criticality. Next, you can give the instructions to log in. Figure 4 gives an image of the page.

You can either give user name and password or give instructions on the flow of the login. Figure 5 shows the login instruction settings page.

Some of the default settings will be taken based on the website and we can change these if we want to. We can select the type of browser to check the application in, and if we want to exclude any paths we can mention that in the settings. Figure 6 shows the crawl settings.

There are some HTTP and advanced settings that we can control according to our requirements. Figure 7 shows the HTTP settings and Figure 8 shows the advanced settings.

After we have selected the required setting, click on 'Scan'. Then the pop-up shown in Figure 9 comes up.

We can select the type of scan based on our requirements. We can schedule scans like instant scan, scheduled scan or a recurring scan.

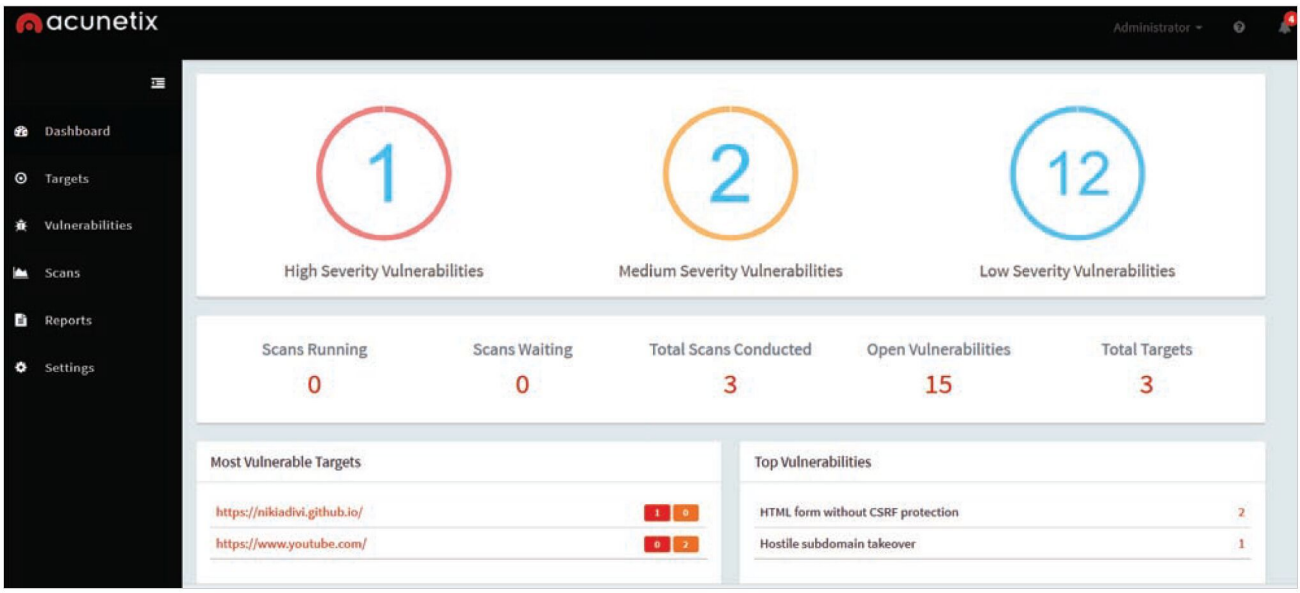



Figure 2: Dashboard of Acunetix

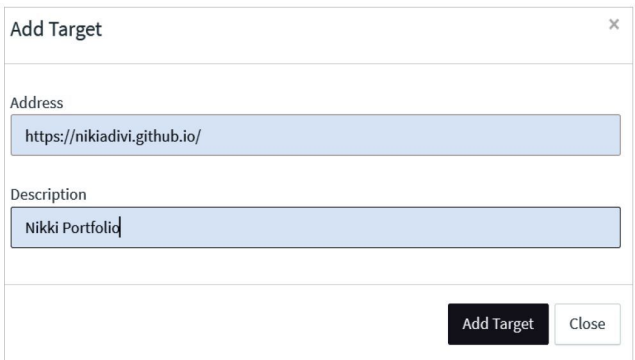



Figure 3: 'Add Target' pop-up

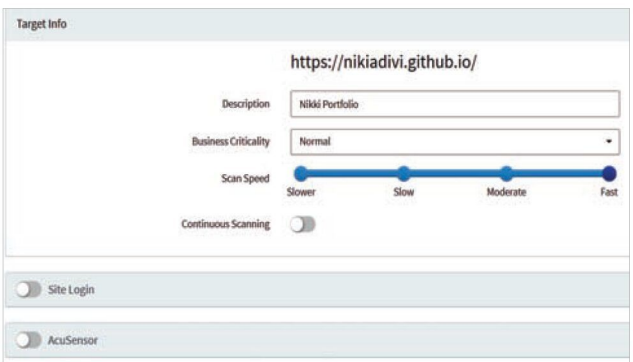



Figure 4: Page after adding *Target*

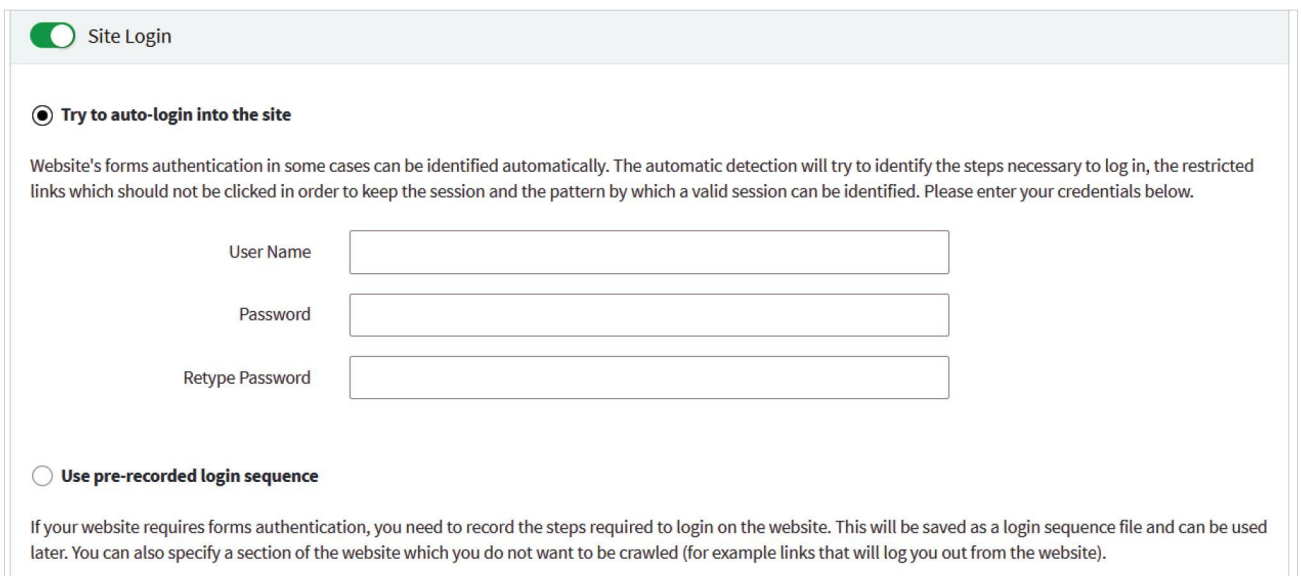



Figure 5: Login instruction settings page

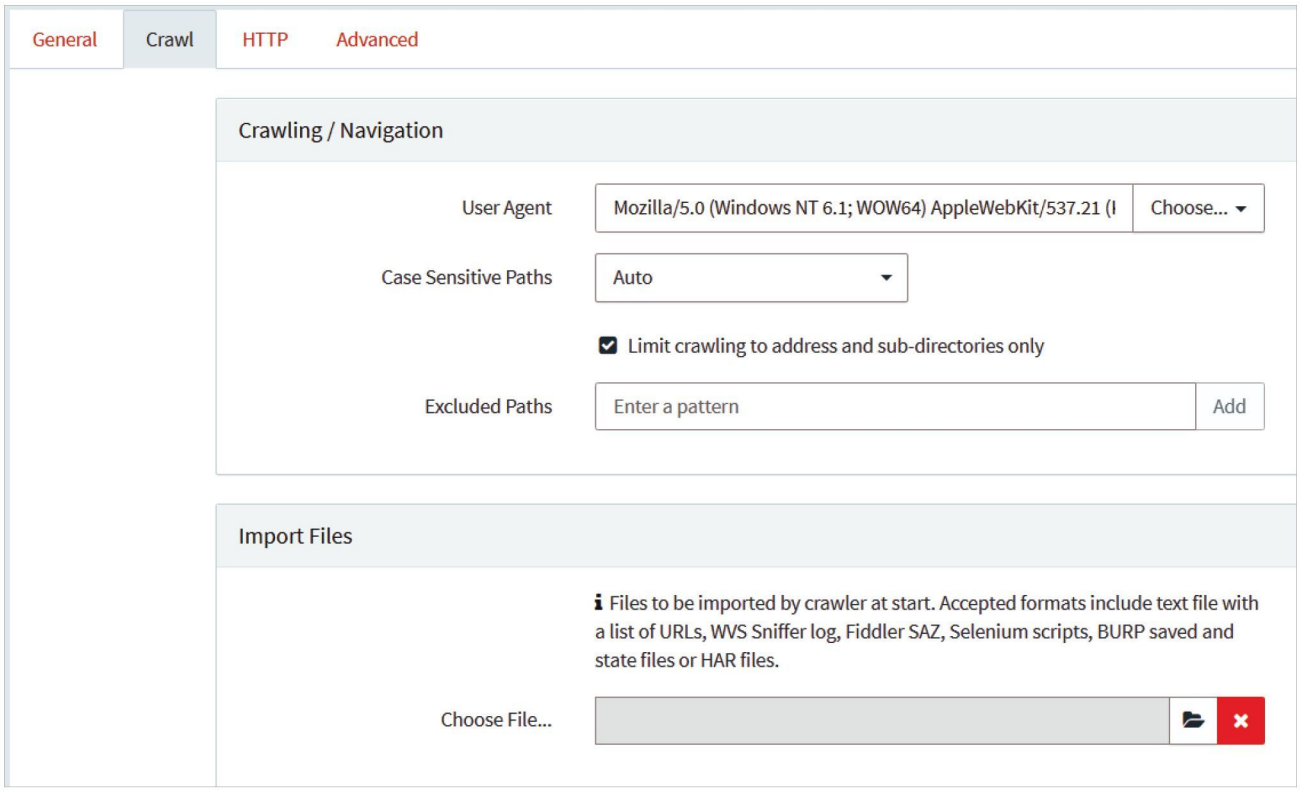



Figure 6: Crawl settings page

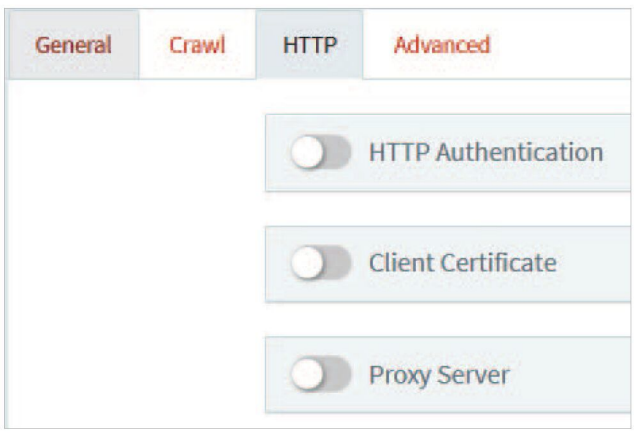



Figure 7: HTTP settings page

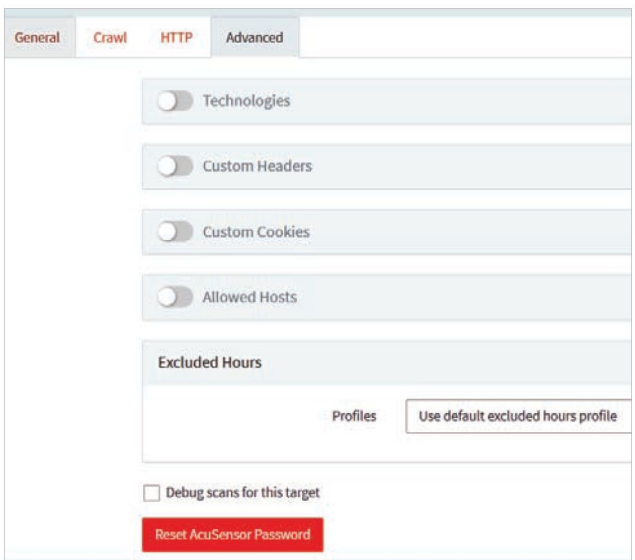



Figure 8: Advanced settings page

After selecting the scan, click on 'Create scan'. Figure 10 shows the report of the scan.

There will be four levels of security, and the applications classify all the vulnerabilities into those levels. You will get the results on completing the scan.

You can select a vulnerability to find out the problem. Figure 11 gives an example of a vulnerability report.

## GuardRails

GuardRails manages the usage of open source and commercial security tools by seamlessly integrating them into your current development workflow. It carefully picks each security rule of the security tools in order to minimise noise and only report high-impact and important security problems. Installing and setting up security technologies may be time-consuming and difficult, even for a single repository. GuardRails streamlines, accelerates, and rewards that process for developers.

It only takes a few minutes to install GuardRails across all your repositories. Once engaged, GuardRails examines all new code updates to look for security flaws before demonstrating to users in detail how to fix them.

GuardRails is different from other solutions in four key areas:

- Version control system integration
- Security tool orchestration
- Security rules curation
- False positive detection

## Configuration of GuardRails

Since it is a web application there is no need to install any

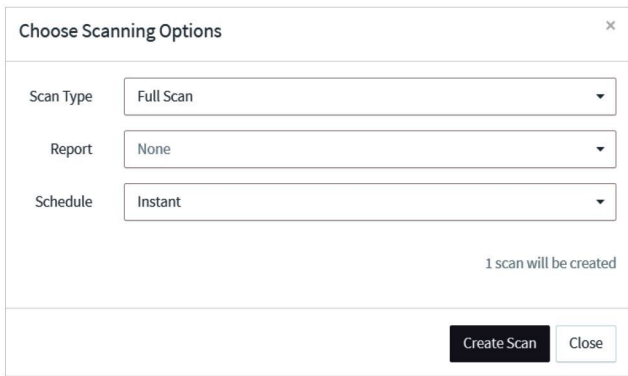



Figure 9: 'Scan' pop-up

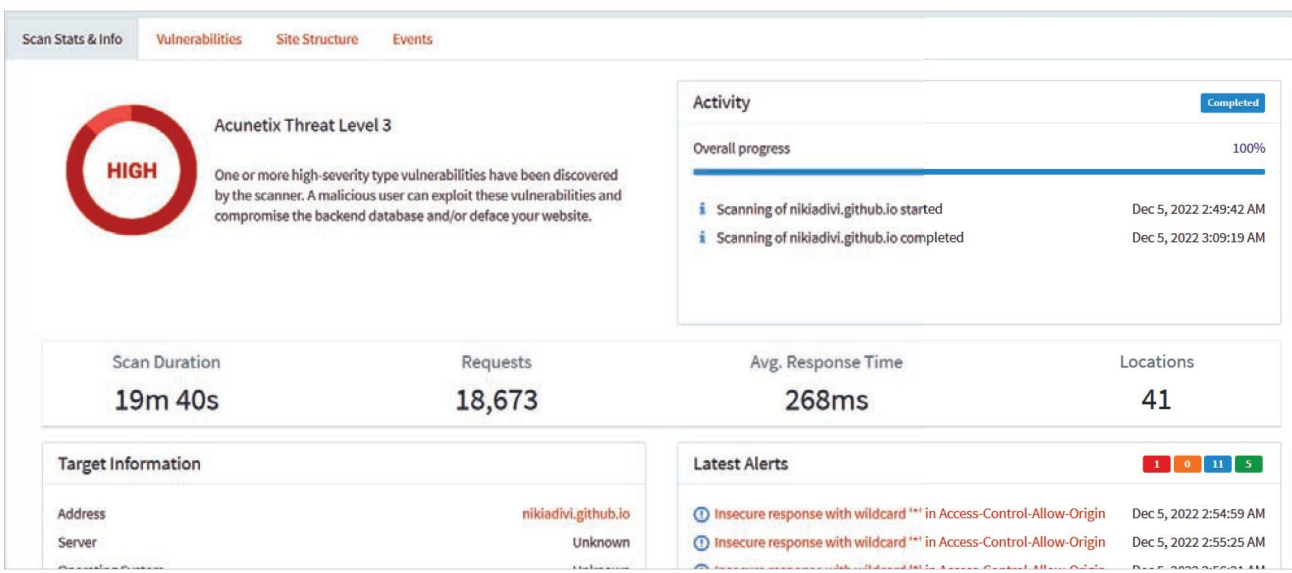



Figure 10: Report of full scan

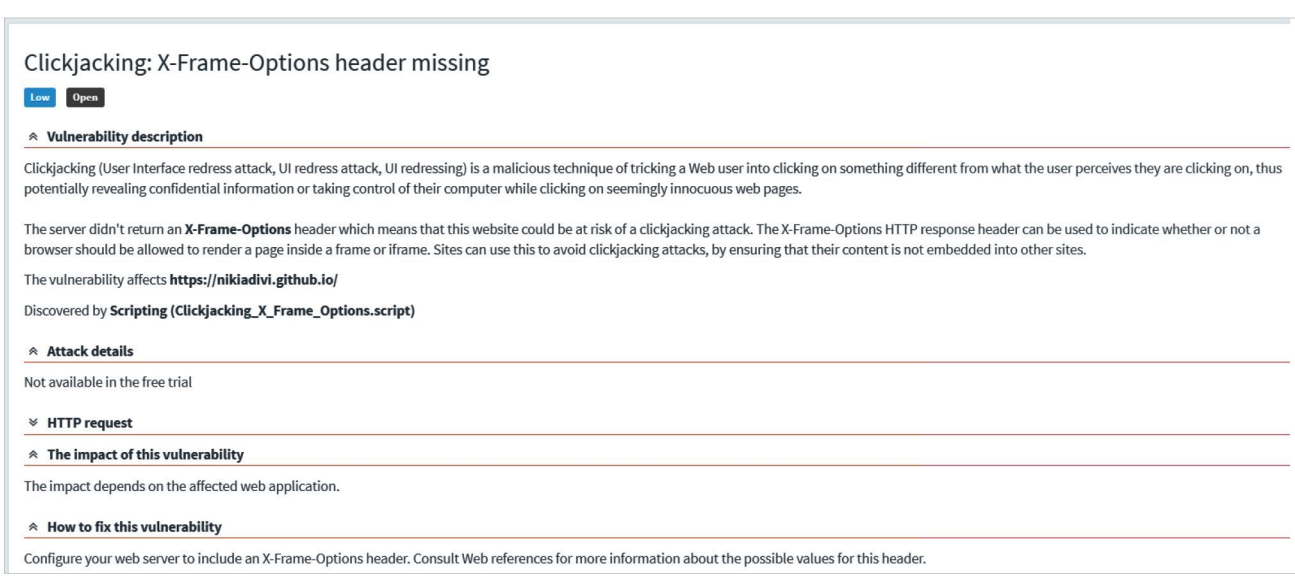



Figure 11: A vulnerability report

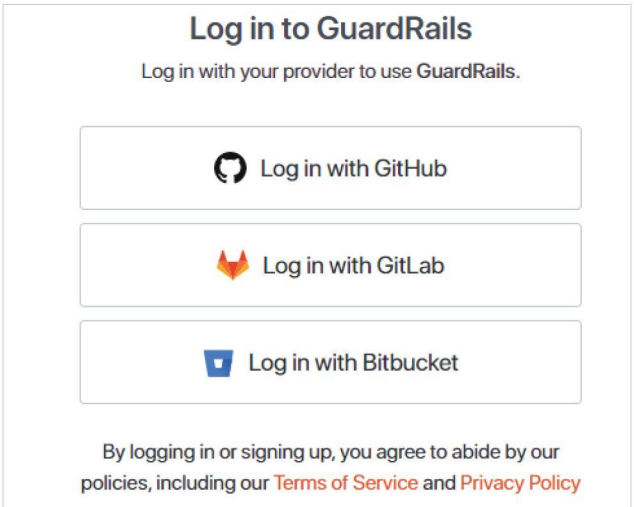



Figure 12: Login page of Guardrails

application on our local computer. Just go to the website and login either with GitHub, GitLab or Bitbucket. The link is *https://dashboard.guardrails.io/login*.

After logging in with your account, give permission from your GitHub or other platforms to access your repo from GuardRails.

Once your repo is added, it will be displayed on the dashboard, as shown in Figure 13.

You can now select the repo you want to scan, and select 'scan' as shown in Figure 14.

After the scan is done, you will get a report with all the vulnerabilities given in three levels. An example report is shown in Figure 15.

You can also select any vulnerability and get the details of where the problem has been identified (Figure 16).

After you verify the problem in the code and rescan it, it won't be shown again if solved.

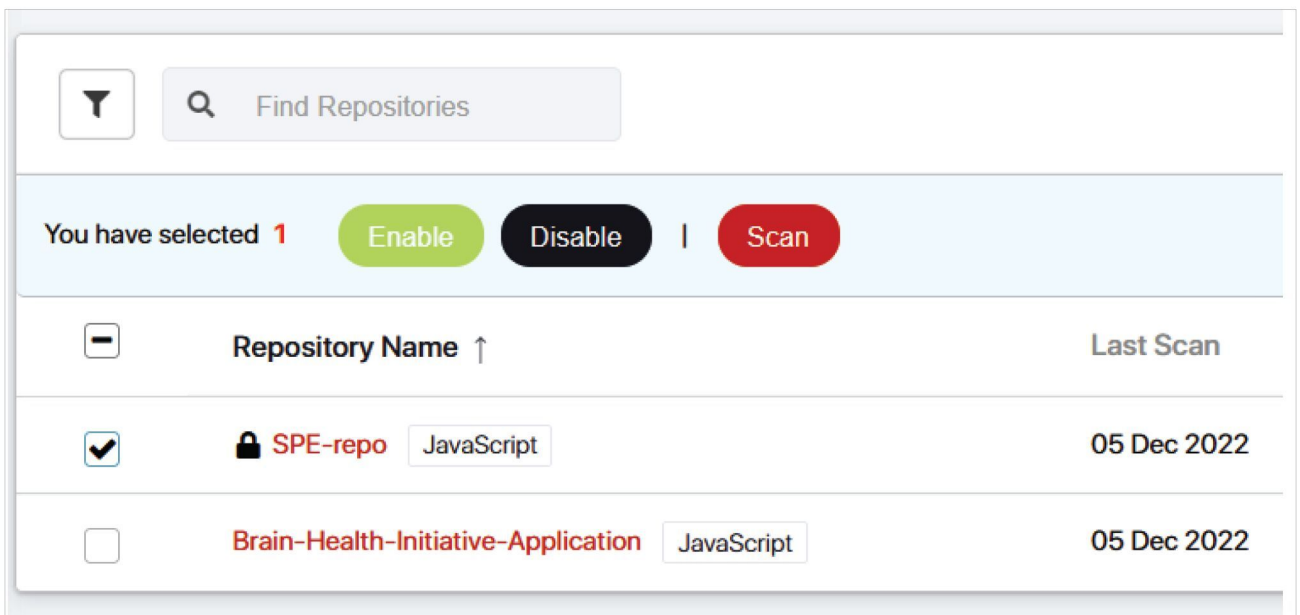



Figure 13: Dashboard page

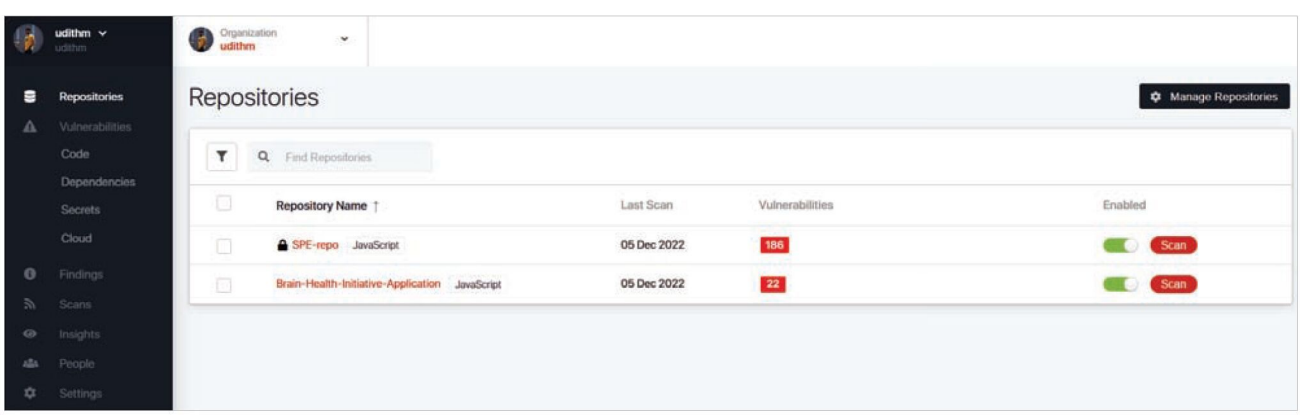



Figure 14: Select the repo to scan

## Nuclei

An upcoming open source DAST tool is Nuclei. It is an application that needs to be downloaded and installed on our local computer.

Using a template, Nuclei is used to deliver queries across targets, producing no false positives and providing rapid scanning on many hosts. Nuclei scans a wide range of protocols, including TCP, DNS, HTTP, SSL, File, Whois, WebSocket, Headless, etc. It enables complex and flexible templating, which may be applied to represent different security checks.

The reference for the nuclei repository is *https://github.com/projectdiscovery/nuclei*.

The security tools covered in this article are efficient and easy to integrate with the development pipeline. Acunetix and Guardrails showcase cutting-edge technologies that are constantly improving due to their intuitive user interfaces, deeper feature sets, quick testing, and enhanced efficiency. In contrast to Guardrails, which is immediately connected to GitHub or other accounts where it is simple to access projects, Acunetix requires installation and a subscription edition for additional

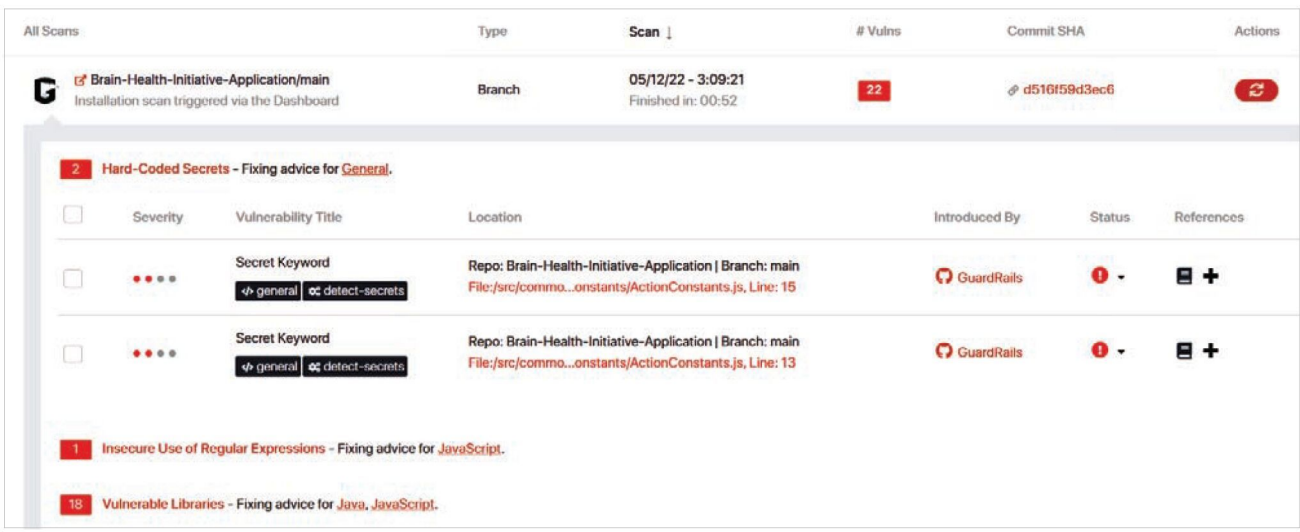



Figure 15: Report of an example scan

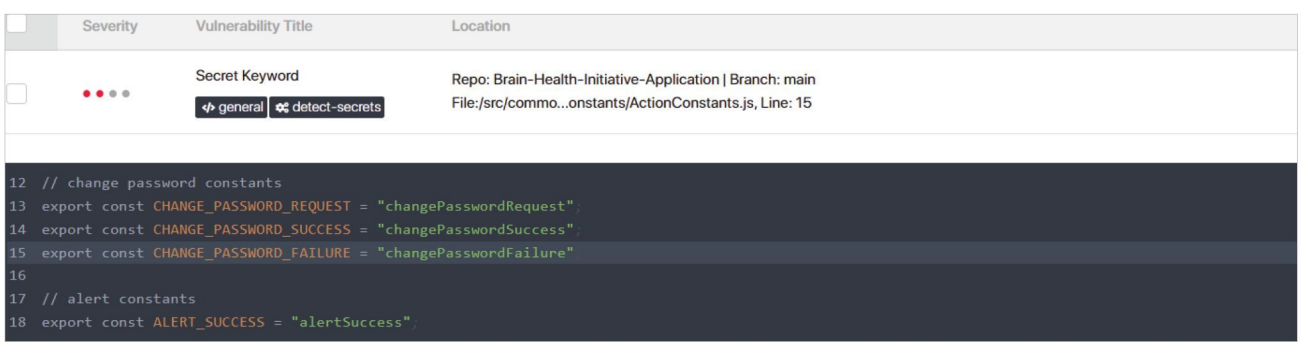



Figure 16: Vulnerability description

capabilities. Nuclei is also being used frequently and helping to fix security related issues in developed software. END

### References


- DevSecOps: Static Application Security Testing Using Snyk and SonarQube, by Shriya Kabra and B. Thangaraju, *Open Source For You*, January 2021, pp. 69-74
- DevSecOps: Integrating a Dynamic Application Security Testing Tool with Jenkins, by Akhilank M.J. Kaipu and B. Thangaraju, *Open Source For You*, January 2021, pp. 84-88
- Acunetix Web Vulnerability Scanner download, available at *https://acunetix-web-vulnerability-scanner.software.informer.com/download/#downloading*
- GuardRails home page at *https://docs.guardrails.io/docs/what-is-guardrails*



**By: Udith Sai M. and Dr B. Thangaraju**

The authors are associated with the Open Source Technology Lab in the International Institute of Information Technology, Bengaluru.

# Worried About Cyber Security? Look for AI and ML Based Solutions

As organisations rely more and more on the internet, they face the risk of complex and evolved cyber threats. Open source AI and ML based cyber security solutions are the need of the hour to counter these threats.


Did you know that 57.6 per cent of the world's population uses social media today, with average daily usage being 2 hours and 27 minutes? There are 5.29 billion unique mobile users across the globe currently, which is equal to two-thirds of the world population. By 2025, the amount of data generated each day is expected to reach 463 exabytes globally.

The rise of internet usage leads to security concerns like cyber attacks and cyber threats. A cyber attack can use malware or ransomware to gain access to data, disrupt digital operations or misuse information. These threats are not only complex but are also difficult to detect. A cyber security practice is required to control these threats and attacks across the organisation.

Cyber security is a technology or process to protect networks, devices, information, programs and data from attacks, damages and unauthorised access. There are many ways to protect data and infrastructure, including intrusion detection, malware protection, and more.

## Classification of cyber security

Cyber security encompasses all the measures taken to protect an entity from cyber threats and secure data. It can be broadly classified into the following distinct security areas:

- Infrastructure security, covering DNS security, mail security, security information and event management
- Application security, which focuses on preventing malware from infecting software and devices
- System security addresses Windows/Linux server security, and

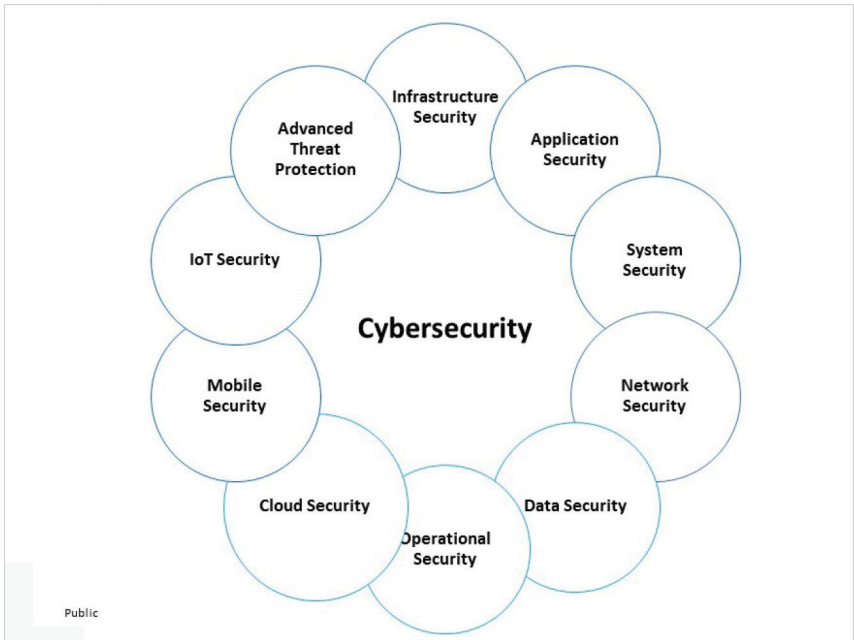



Figure 1: Types of cyber security

vulnerability and patch management
- Network security safeguards a computer network from intruders
- Data security, both in storage and in transport, which uses automated data encryption and data leakage prevention techniques
- Operational security refers to the procedures and actions used to manage and secure digital assets
- Cloud security needs to be continuously monitored and updated to safeguard data from attacks
- Mobile security helps in authentication and on-boarding, rogue access point detection, and managing wireless secure protocols
- Internet of Things (IoT) security protects products and services against threats
- Advanced threat protection covers botnet protection, malware analysis and anti-malware solutions, forensic solutions and automated security analysis

## Industry trends in cyber security

According to a Grandview research report, the artificial intelligence component of the global cyber security market reached US$ 13.29 billion in 2021. It is expected to grow at a compound annual growth rate of 24.3 per cent from 2022 to 2030 to reach US$ 93.75 billion by 2030.

Mobile banking malware or assaults are on the rise (by 50 per cent), making portable devices a target for hackers. All our photographs, financial transactions, emails, and interactions put us in danger.

Another industry report says that organisations will be increasingly afraid to stack their security measures in 2023 because of these cyber security developments. They are likely to spend more than ever on asset protection this year, with estimates of US$ 100 billion or more.

Some key players operating in artificial intelligence in the cyber security market include:
- Acalvio Technologies, Inc.
- Amazon Web Services, Inc.
- Cylance Inc. (BlackBerry)
- Darktrace
- FireEye, Inc.
- Fortinet, Inc.
- IBM Corporation
- Intel Corporation
- LexisNexis
- Micron Technology, Inc.

## Core concepts of AI and ML in cyber security

Artificial intelligence (AI) is a simulation of the human intelligence process by computers. It helps in identifying vulnerabilities, threats, and attacks in cyber space. This is the umbrella discipline under which fall machine learning and deep learning.

Machine learning (ML) uses existing behaviour patterns, making decisions based on past data and conclusions. ML consists of the programs developed to access the data stored on the system and make it more intelligent by studying the patterns and providing better support through continuous learning. There are three types of ML algorithms.
- ***Supervised learning:*** In this system both the input and the desired output data are provided. Input and output data are labelled for classification to provide a learning basis for future data processing. The term supervised learning comes from the idea that an algorithm is learning from a training data set, which can be thought of as the teacher. Data sets are labelled so that patterns can be detected and used to label new sets.
- ***Unsupervised learning:*** This involves the training of an AI algorithm using information that is neither classified nor labelled, and allowing the algorithm to act on that information without guidance. Data sets aren't labelled and are sorted according to similarities or differences.
- ***Reinforcement learning:*** This training method is based on rewarding desired behaviours and/or punishing undesired ones. Data sets aren't labelled but, after performing an action or several actions, the AI system is given feedback.

Deep learning (DL) is a type of ML algorithm that uses neural networks (NN), a modelling approach inspired by how our brains work. It comprises millions of neurons, connected to each other, organised in hundreds or more layers.

To address complex cyber attacks and threats, several organisations are implementing AI and ML based security solutions and technologies.

AI and ML have evolved as an optimal solution in cyber security, with solution techniques/algorithms being applied across all market sectors. They play a crucial role in the development of automated security systems, natural language processing, face recognition, and autonomous threat detection.

## Popular AI/ML approaches to cyber security

Some of the most prominent and popular ways in which AI and ML detect cyber threats are described below briefly.

***Malware detection and identification:*** In this approach, AI and ML algorithms help in identifying malicious files and filter them before they reach the end user. Many different AI and ML approaches have been used to detect malware. These solutions can detect, respond, and remediate in real-time. Most prominent among these are:

- Machine learning and data mining to look for malware source code repositories using a technique called 'SourceFinder' and analyse them based on characteristics and properties
- Machine learning to look for a particular string within files that could indicate the presence of malware or malicious code and classify them
- Usage of AI/ML to detect patterns in binary executable files and determine if they are malicious
- Utilising visual binary patterns identified in the code and a type of self-organising network that adapts over time

***Behavioural threat analysis:*** Threat analysis is a cyber security strategy that aims to assess an organisation's security protocols, processes and procedures to identify threats, vulnerabilities, and gather knowledge of a potential attack in advance. A threat analysis consists of the information and assets that need to be protected in terms of confidentiality, integrity, and availability. Some of the techniques used to identify the threats are:

- ML techniques called user and event behavioural analytics (UEBA) to analyse and recognise typical behaviours and patterns in user accounts and endpoints. These can detect security incidents that violate predefined operational rules, employ novel attack patterns, or span multiple organisational systems and data sources.
- Semi-supervised learning is another approach used for threat detection.

***Spam detection:*** Spam emails may have inappropriate contents, links and attachments, which can lead to security issues. Uninvited bulk emails generally belong to the category of spam. AI and ML models can be used to detect spam by analysing the content of the mail/message and looking for patterns.

AI is able to detect that messages are spam without requiring any human intervention. Some of the spam detection algorithms are:

- Bayes algorithm that helps to filter out some spam emails
- AI and ML based spam detection unsupervised text mining model used to detect the possibility of false reviews

***Network intrusion detection:*** AI and ML based intrusion detection systems (IDS) develop intelligent systems to detect, classify, and respond to cyber attacks. IDSs identify malicious behaviour and stop it before it causes any damage.

An IDS is usually configured with a set of rules such as use of certain words in the subject line of an email message or sending of too many messages in a given time period. It helps to generate alerts when it detects an event related to an attack or intrusion attempt.

IDS can be implemented as a standalone system or as an add-on module to security software — for example, antivirus programs.

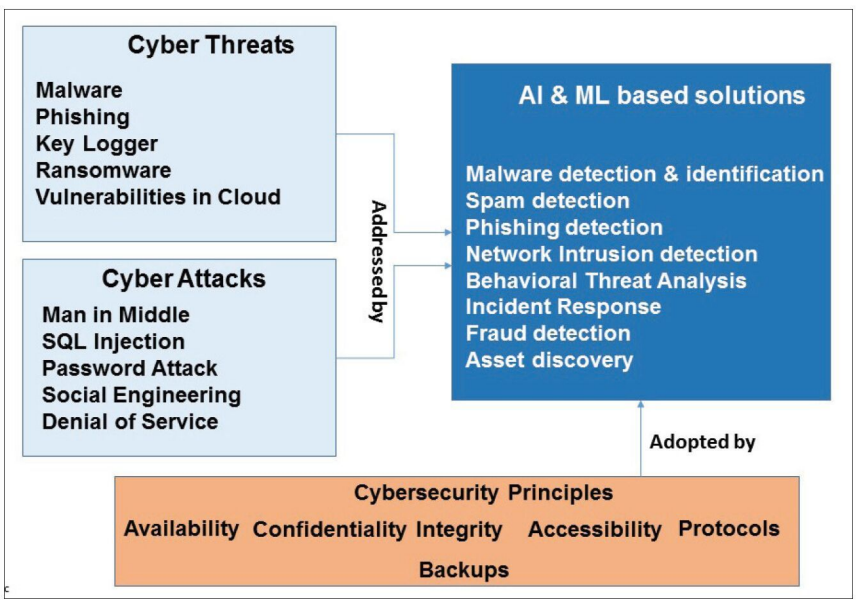



Figure 2: AI and ML based cyber security solutions

IDS responses can be categorised into two main types:

- *Active defence:* An active defence is 'real-time' defence where the system initiates an action at the moment of detection rather than waiting for a report from another system.
- *Passive defence:* In this type of defence, IDS only responds after receiving and processing information that an intrusion attempt has taken place.

The goal of an AI and ML based algorithm is to optimise certain features and improve its classifiers, so that it reduces the number of false alarms that come up while trying to identify an intruder.

***Phishing detection system*:** An AI based system detects phishing emails by analysing the content of the email and comparing it with a database of known phishing emails. The phishing detection system (PDS) can also detect if the sender is spoofing another person's identity.

PDS can also be used with voice, video, and image messages. The system activates when users receive a suspicious email or when they send an email containing personal information. Some of the features of the PDS are:

- Automatically detects email phishing scams
- Stores emails with malicious content in a quarantine folder
- Triggers user notifications when the system detects a new virus in an email
- Maintains detailed logs of all email activity
- Detects emails that contain phishing links
- Automatically generates a report of every detected email

The goal of the PDS is to automatically detect and report emails that contain phishing links.

***Automated processes that optimise human analysis:*** Automated processes can be set up by analysing reports on past actions generated by security analysts to identify and respond to certain attacks successfully. AI algorithms use this knowledge to build a model, which can be used later for identifying similar cyber activities. Using this model, AI algorithms

| Feature | Open source tool/product | Remarks |
|---|---|---|
| Malware detection and identification | REMnux, OpenEDR | **REMnux** is a free Linux toolkit for reverse engineering and analysing malware. **OpenEDR** helps organisations to secure their infrastructure against malware, ransomware, data breaches and other threats. |
| Threat analysis | Nmap, Metasploit | **Nmap** provides methods to find open ports, detect host devices, verify active network services, fingerprint operating systems and locate potential backdoors. **Metasploit** helps security professionals perform simulation attacks to find loopholes in a system. |
| Spam detection | FortiClient | **FortiClient** reduces the risk of malware, and blocks spam URLs as well as exploit kits. |
| Network intrusion detection | Security Onion, Snort, PfSense, OSSEC | **Security Onion** provides network monitoring via full packet capture, host-based and network-based intrusion detection systems, log indexing, and search and data visualisation features. **Snort** is capable of real-time traffic analysis and logging. **PfSense** is configured for intrusion detection and prevention, traffic shaping, load balancing and content filtering. |
| Phishing detection | Gophish | **Gophish** provides a full-featured toolkit for security administrators to build their own phishing campaigns. |
| Asset discovery | AlienVault, OSSIM | **OSSIM** offers end-to-end security information and event management through asset discovery, behavioural monitoring, and event correlation. |
| Incident response | OpenVAS, Nikto | **OpenVAS** is an all-in-one vulnerability scanner. It tests for security issues, misconfigured systems and outdated software. |
| SQL injection flaw detection | Sqlmap | **Sqlmap** automates detecting and exploiting SQL injection flaws of database servers, enabling a remote hacker to take control. |

Table 1: Popular open source AI and ML based cyber security tools

respond to attacks without human interference.

***Incident response system:*** AI and ML based systems help in providing incident responses, enabling organisations to manage security alerts appropriately. AI automated incident responses mitigate vulnerabilities and deliver faster responses to such events.

***Fraud detection:*** AI and ML based systems can be used to create models to recognise fraud-related patterns. As more data is fed to the system, the AI model becomes more accurate.

***Asset discovery:*** AI and ML can be used for automating the discovery of all key devices and applications. This can play a huge role in mitigating risks.

## Key open source AI and ML cyber security tools

Open source cyber security tools help organisations to protect their devices, data, and user landscapes from internal and external threats. These tools can be proactive or reactive, allowing organisations to test systems and check for vulnerabilities or monitor active systems to pre-empt incoming attacks.

Cyber security tools have the following important features:

- Business-need alignment and organisation readiness
- Highly scalable
- Support heterogeneous environments and can adapt easily
- Have proper industry support for technology
- Integrate easily with an organisation's systems and tools

Table 1 highlights the most popular and important open source AI and ML based cyber security tools.

## Benefits of AI and ML in cyber security

Integrating AI and ML into a cyber security system has quite a few benefits:

- Ability to detect nuanced attacks, strengthen security, and enhance incident response
- Improves the detection and response cycle time
- Organisations can rapidly quantify risks and accelerate analyst decision-making with data-driven mitigation measures
- Prevents and mitigates cyber security breaches and malicious attacks
- Improved workforce experience
- Improved customer satisfaction and brand reputation due to heightened cyber security protection and increased trust in the organisation's security protocols

Cyber threats have become innovative and are constantly evolving. Also, data is filled with new patterns that are hard to capture and analyse manually. AI and ML offer a powerful way of identifying vulnerabilities, threats and attacks across organisations as well as social media. AI and ML algorithms help to detect and analyse enormous amounts of data, and the solutions they offer are more robust, flexible, and scalable.

The main targets of AI and ML based algorithms for cyber security are malware detection, network intrusion detection, and phishing and spam detection. Some of the major adopters of AI and ML based cyber security solutions are Google, IBM, Juniper Networks, Apple, Amazon, and Balbix. More and more companies are joining this bandwagon.

To sum up, integrating AI and ML into your cyber security solutions today is not an option but a necessity if you want to counter the emerging complex security threats.


## *Acknowledgements*

Dr Behara would like to thank Santosh Shinde of BTIS, Enterprise Architecture division of HCL Technologies Ltd for giving the required time and support in many ways when this article was being written as part of Architecture Practice efforts.


END


**By: Dr Gopala Krishna Behara and Raja Sekhar Amirapu**

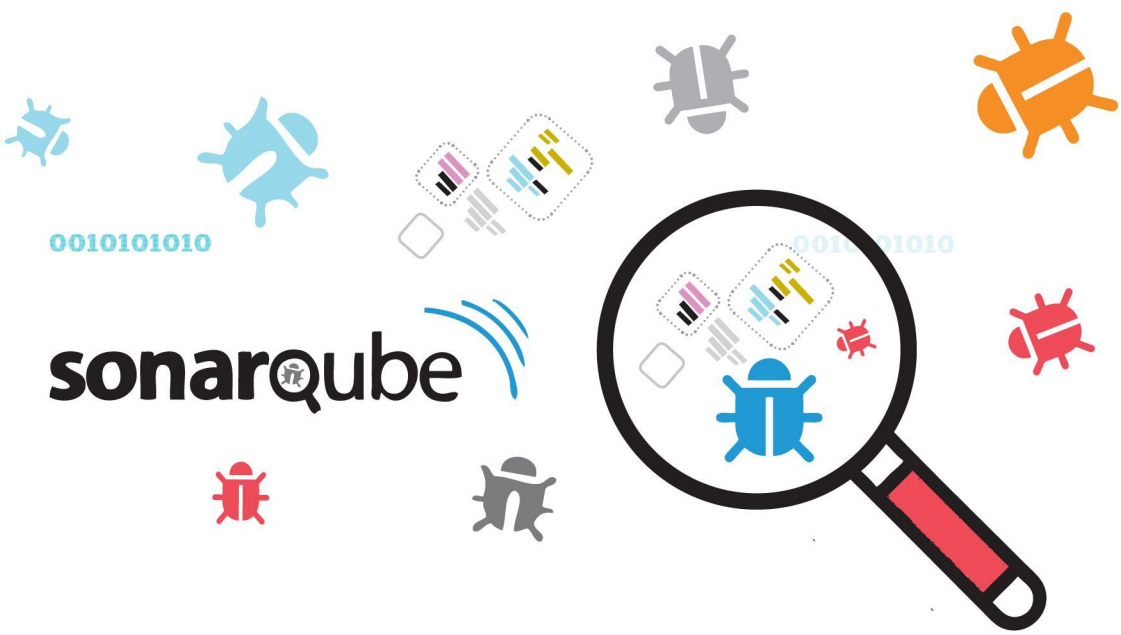
**Dr Gopala Krishna Behara** is an enterprise architect in the BTIS Enterprise Architecture division of HCL Technologies Ltd. He has 'a total of' 27 years of experience in the IT industry.

**Raja Sekhar Amirapu** is a senior architect at Tech Mahindra. He has 'a total of' 26 years of experience in the IT industry.


***Disclaimer:*** *The views expressed in this article are that of the authors and HCL or Tech Mahindra do not subscribe to the substance, veracity or truthfulness of the said opinion.*

# Static Application Security Testing (SAST) with SonarQube

SAST stands for static application security testing. It focuses on analysing the source code of an application to identify bugs, security vulnerabilities and code smells. The objective of SAST is to identify these issues early in the software development life cycle before they are identified and exploited in the production environment. SonarQube, a popular open source tool, can help with this.




SAST usually analyses an application's source code, configuration files, infrastructure configuration and build scripts to identify potential bugs and vulnerabilities. We don't need to execute the code to analyse it in SAST. This helps to? It helps to reduce the risk of security incidents and ensure that the application is secure before it is deployed in a production environment.

SAST helps to identify potential security risks such as:

- SQL injection attacks
- Cross-site scripting (XSS) attacks

SAST tools usually use a combination of rule-based analysis and code instrumentation to identify security risks and report them. SAST is often used with other security testing techniques popularly known as dynamic application security testing (DAST) and penetration testing (pen testing). We can also automate the process of code analysis to identify bugs, vulnerabilities and code smells to deliver good quality applications with speed integrated in them.

SAST is typically managed and monitored by development teams. It can be integrated in the continuous integration and continuous delivery (CI/CD) pipeline to maintain process and stability, with quality gates integrated into it.

## Overview of SonarQube

SonarQube is a very popular open source tool for continuous inspection of code quality. It provides an efficient way to identify and fix bugs, security vulnerabilities and code smells in analysed applications. SonarQube supports multiple programming languages such as Java, Python, Go, C#, and JavaScript. It is very easy to integrate SonarQube with popular CI/CD tools such as Jenkins, Azure DevOps, and GitLab. It also provides a centralised dashboard where you can get

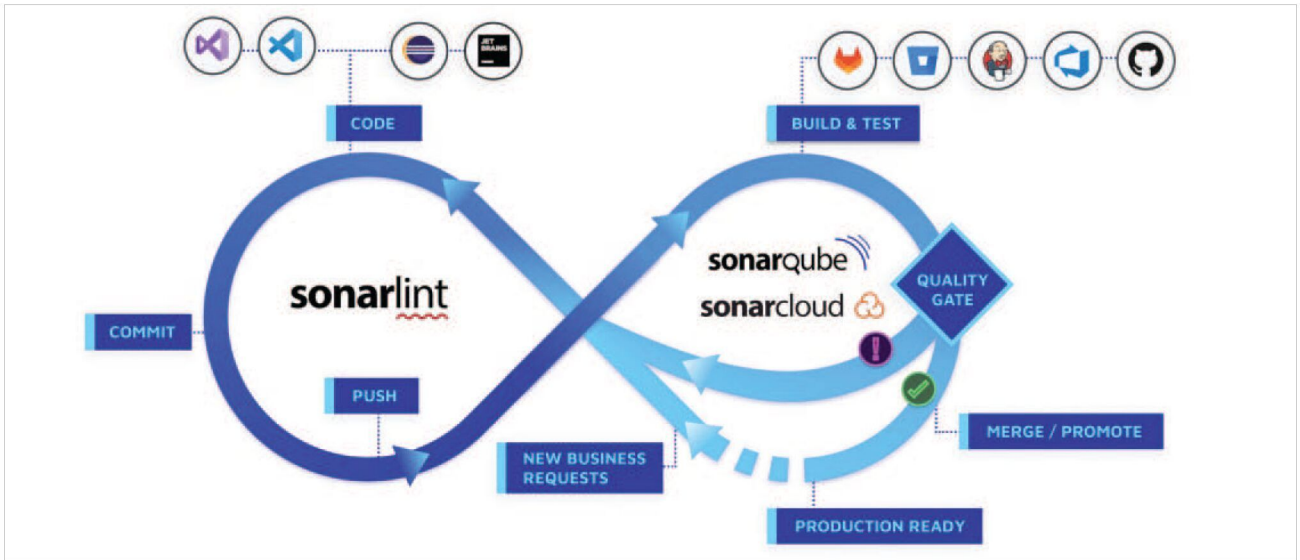



Figure 1: Developing with Sonar (Image: *https://docs.sonarqube.org/latest//*

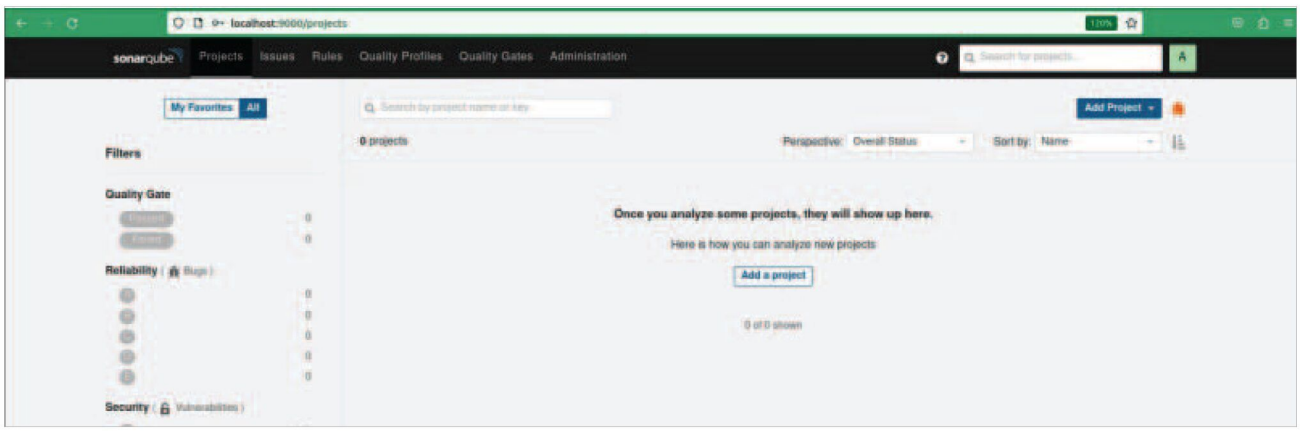



Figure 2: SonarQube dashboard

details of code quality and technical debt. The important thing is that it performs static code analysis. It analyses the source code of an application without running it.

Additionally, SonarQube provides a comprehensive set of reports and metrics for tracking and monitoring code quality over time. These metrics include code complexity, code duplication, test coverage, and so on. SonarQube also offers plugin architecture. Overall, SonarQube is a valuable tool for any development team that wants to improve the code quality and maintainability of its codebase. By automating code inspection by integrating it with CI/CD tools, SonarQube can help teams to deliver better applications faster and with fewer issues.

## How to install SonarQube

Here are the basic steps to install SonarQube on a Windows or Linux machine:

1. Download the latest version of SonarQube. Go to the SonarQube official website *https://www.sonarsource.com/open-source-editions/* and download the latest version of the community edition from *https://www.sonarsource.com/products/sonarqube/downloads/*.
2. SonarQube requires Java 8 or higher to be installed. Download Java from the official Oracle website.

```
osfy@ubuntu:~/Desktop/OSFY/March2023/#tools/
sonarqube-8.9.10.61524/bin/linux-x86-64$ java --version
openjdk 11.0.17 2022-10-18
OpenJDK Runtime Environment (build 11.0.17+8-post-Ubuntu-
1ubuntu222.04)
OpenJDK 64-Bit Server VM (build 11.0.17+8-post-Ubuntu-
1ubuntu222.04, mixed mode, sharing)
```

3. SonarQube requires a database to store its data, so install one. You can use either a MySQL, PostgreSQL, or Microsoft SQL server. Remember that an embedded

database should be used for evaluation purposes only.

4. Configure the database.
5. Extract the downloaded SonarQube archive to a directory of your choice.
6. Start the SonarQube server. Navigate to the directory where you extracted the archive and run the following command.

*Linux:*

```
sonarqube-8.9.10.61524/bin/linux-x86-64$ sh sonar.sh start
osfy@ubuntu:~/Desktop/OSFY/March2023/#tools/
sonarqube-8.9.10.61524/bin/linux-x86-64$ sh sonar.sh start
Starting SonarQube...
Started SonarQube.
```

*Windows:*

```
sonarqube-8.9.10.61524/bin/windows-x86-64/StartSonar.bat
```

7. Open a web browser and navigate to *http://localhost:9000*. You should see the SonarQube login page.
8. Log in to SonarQube. The default login credentials are *admin* for the user name and for the password. You can change the credentials after logging in for the first time. That's it! You have now installed and started SonarQube.

## Quality gates and quality profiles

In SonarQube, quality gates and quality profiles are used to define the acceptance criteria for code quality.

***Quality gates:*** This is a set of predefined conditions that must be met before code can be considered good to go. For example, a quality gate may require that the code has a certain level of code coverage (such as 80 per cent), or that all vulnerabilities have to be addressed. If a quality gate is not met, the CI/CD pipeline must fail, and the development team must address the issues before it can go ahead.

***Quality profiles:*** This is a collection of rules that define the expected quality standards.

Quality profiles can be customised to meet the specific needs of a project or organisation, such as rules can be activated or deactivated or configured as false positive.

By using quality gates and quality profiles, SonarQube makes it easy for teams to enforce best practices and maintain a high level of code quality. The results of the code analysis are automatically evaluated against the quality gates and profiles, ensuring that the code meets the standards set by the quality team and the CI/CD pipeline considers the outcome for next stage execution.

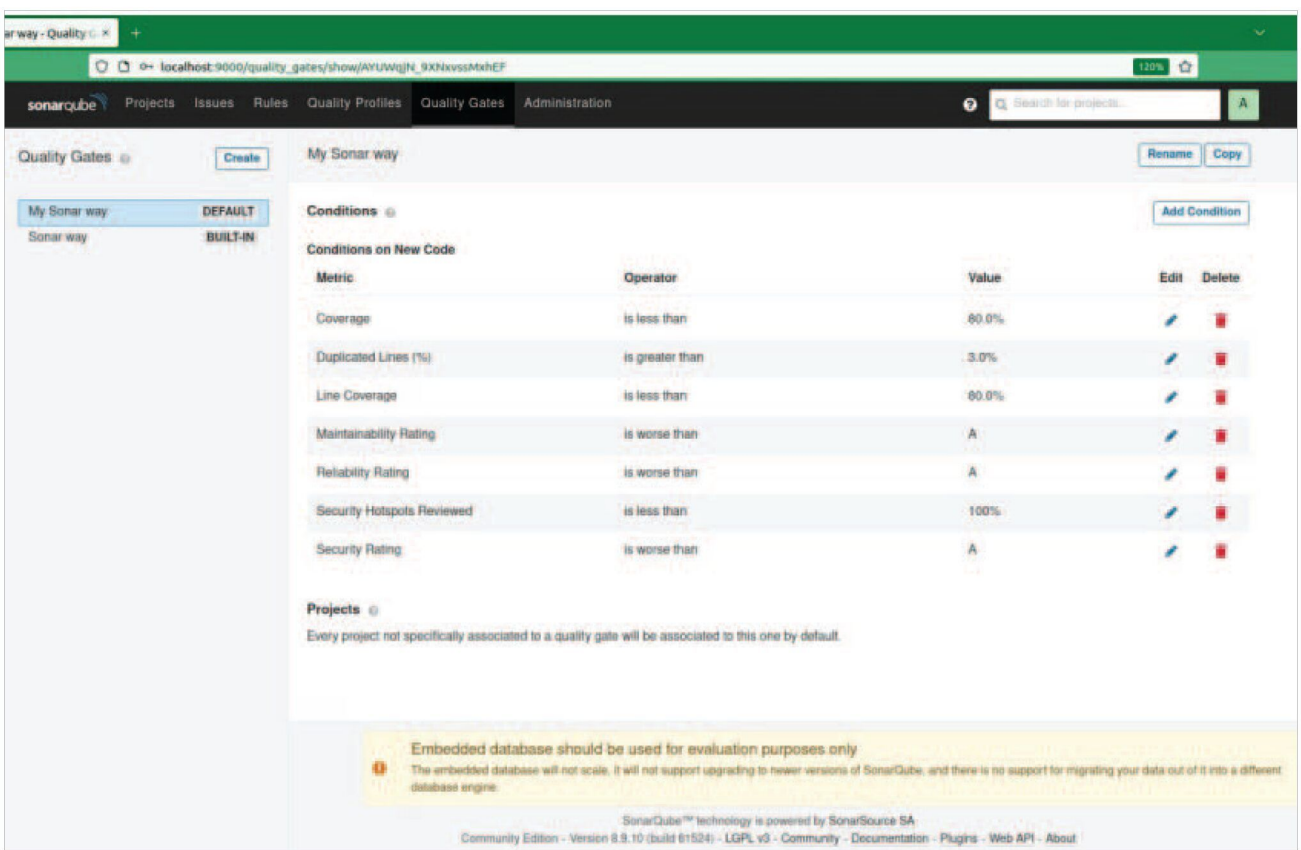



Figure 3: SonarQube quality gate

## Analyse an application using SonarQube

Download SonarScanner from *https://docs.sonarqube.org/latest/analyzing-source-code/scanners/sonarscanner/* based on your OS. The SonarScanner is the scanner to use when there is no specific scanner for your build system.

Create a user token. Let's start analysing the code and monitor code quality.

Go to your sample application directory. We are going to analyse the code using the command line.

```
/home/osfy/Desktop/OSFY/March2023/#tools/sonar-scanner-
4.7.0.2747-linux/bin/sonar-scanner -Dsonar.login=a121ac76da
7addb7392f3efa77347c2561a26e08 -Dsonar.source=src -Dsonar.
projectKey=springtest-app-demo -Dsonar.java.binaries=.
```

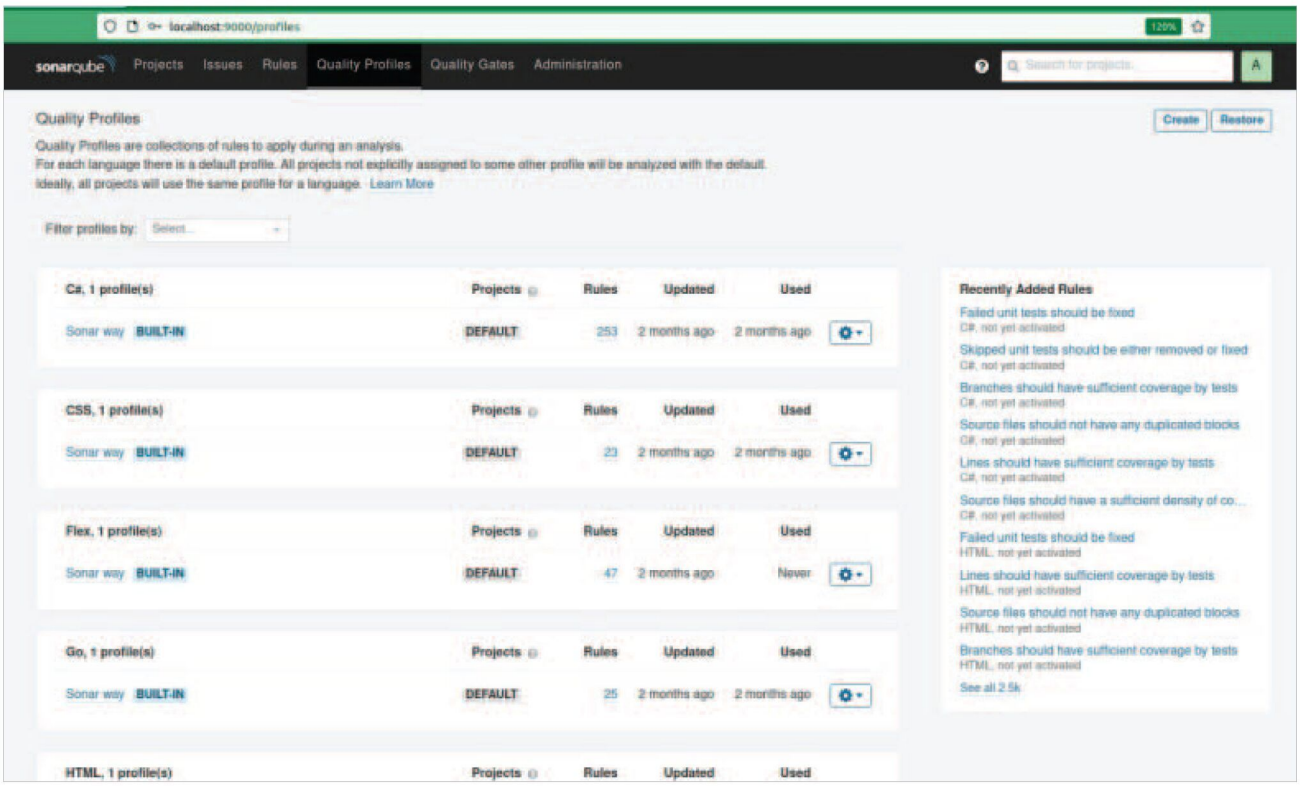



Figure 4: SonarQube quality profile

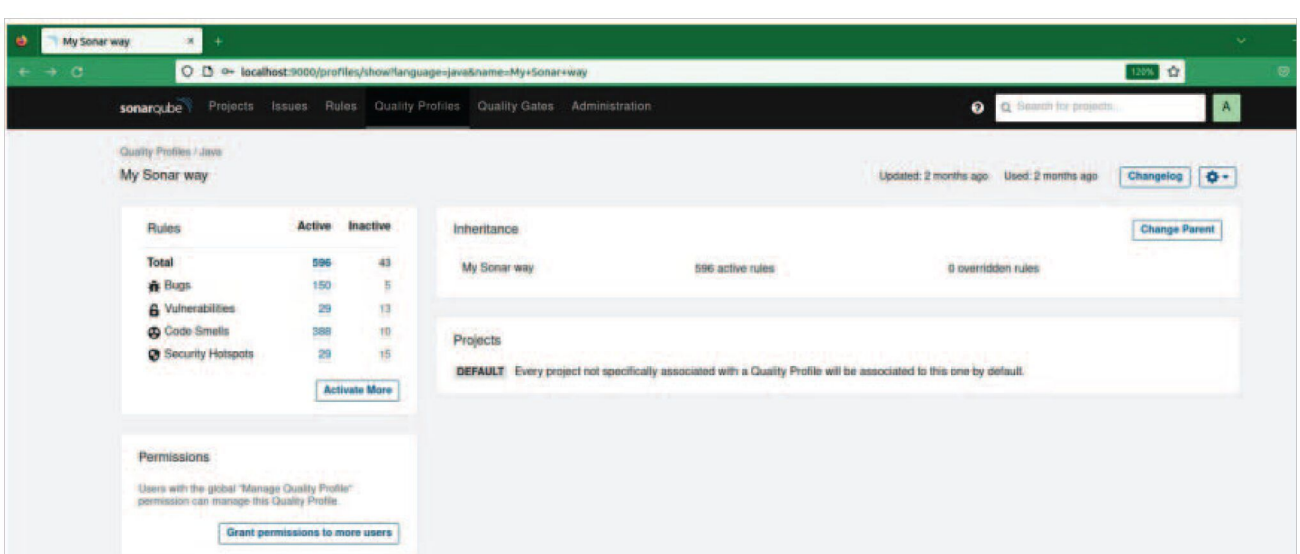



Figure 5: SonarQube quality profile rules

You can also create a configuration file in your project's root directory called sonar-project.properties for configuring parameters as given at *https://docs.sonarqube.org/latest/analyzing-source-code/scanners/sonarscanner/*.

Scroll down in the Terminal.

## Benefits of SonarQube

SonarQube offers several benefits to development teams, such as:

- Improved code quality.
- Increased efficiency as it reduces the time and effort required to identify and resolve code quality issues – it provides solutions also to write code with better quality.
- Better collaboration with the use of a centralised dashboard.
- Increased transparency due to a set of reports and metrics provided by SonarQube.

Improved security by identifying potential security risks early in the development life cycle.

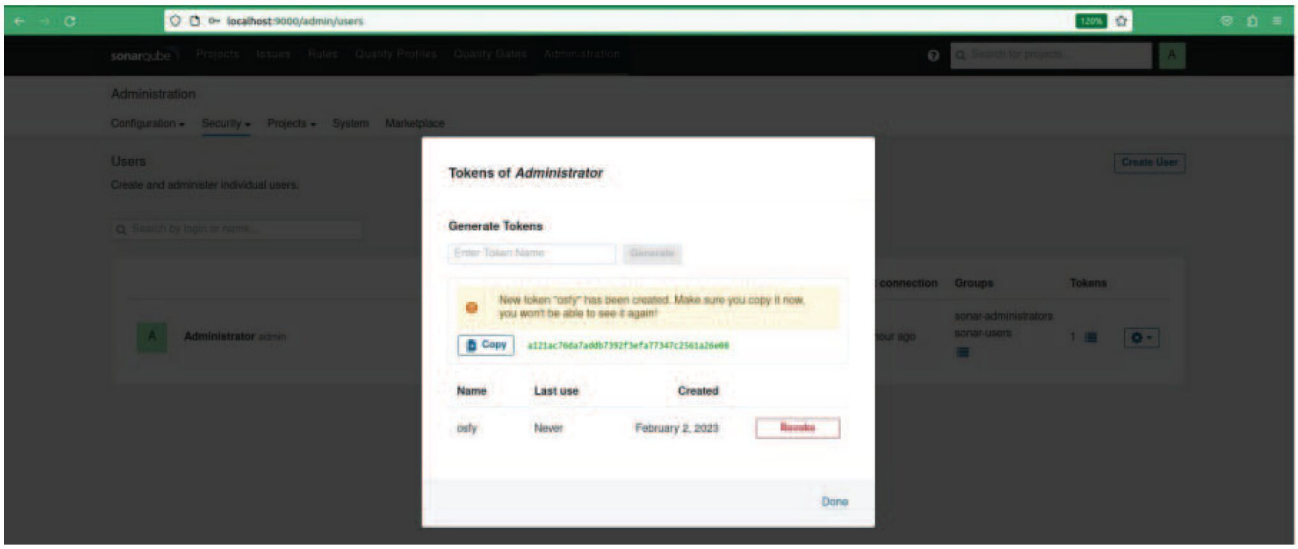



Figure 6: User token

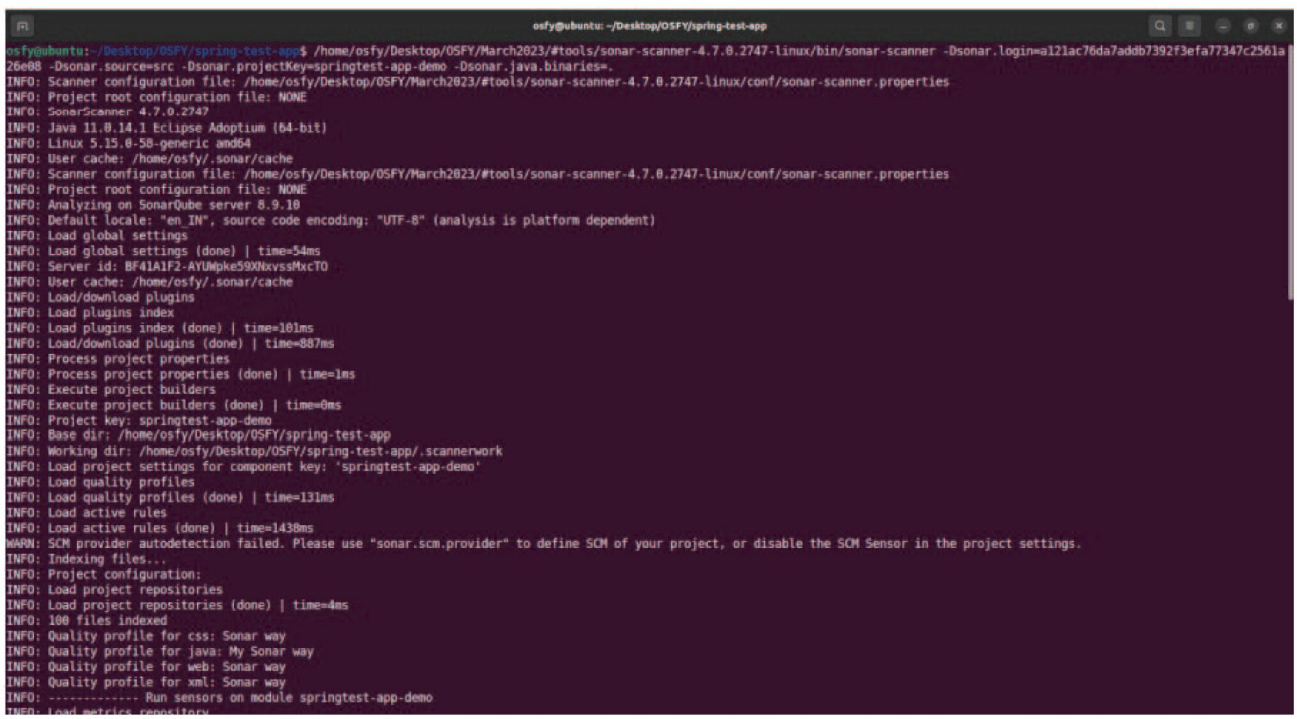



Figure 7: Code analysis from command line

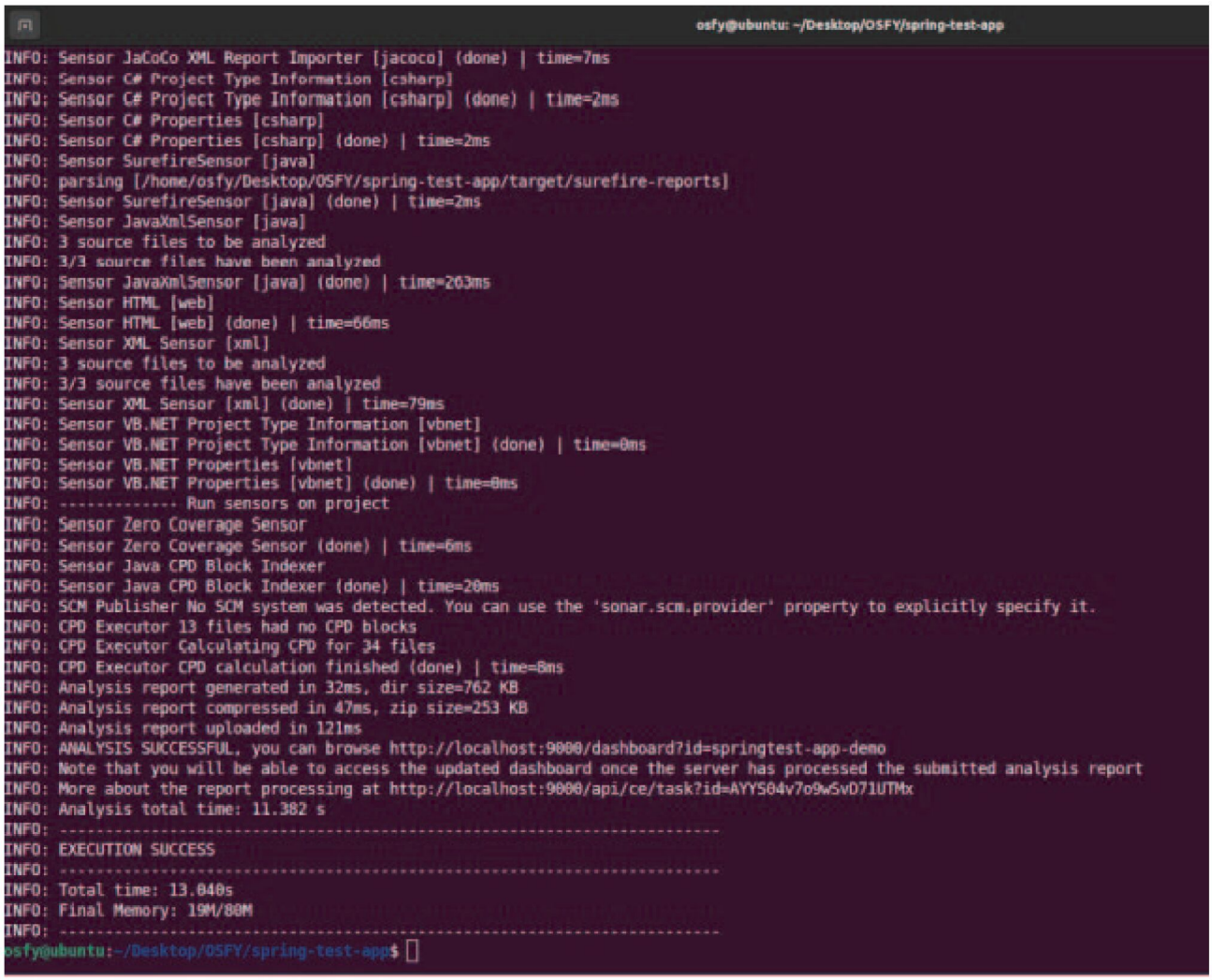



Figure 8: Successful code analysis in SonarQube

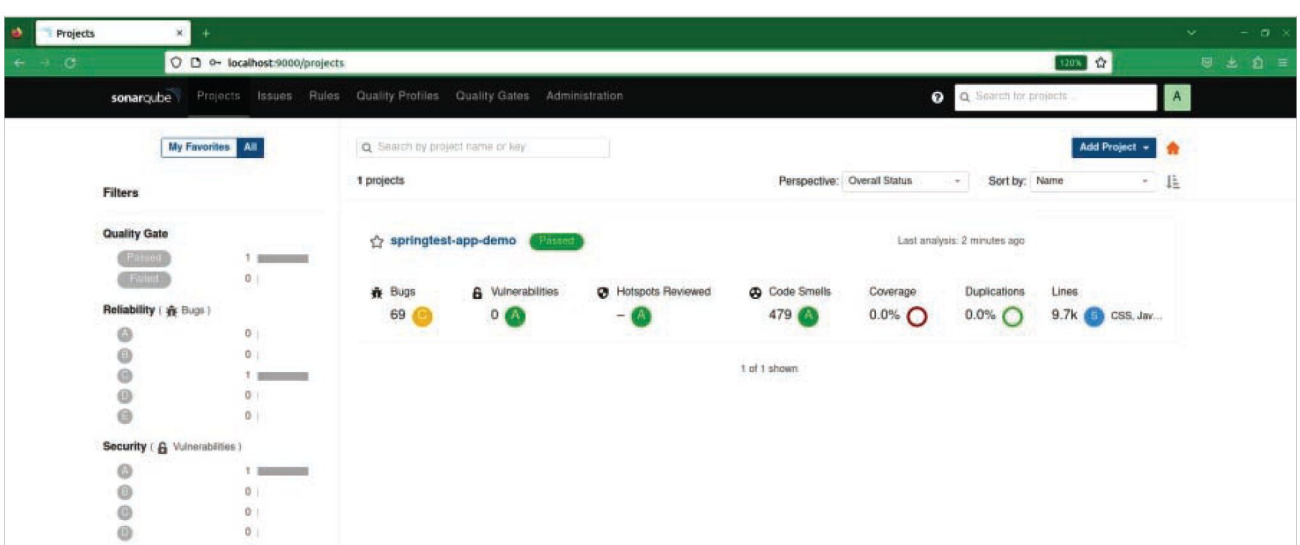



Figure 9: SonarQube summary

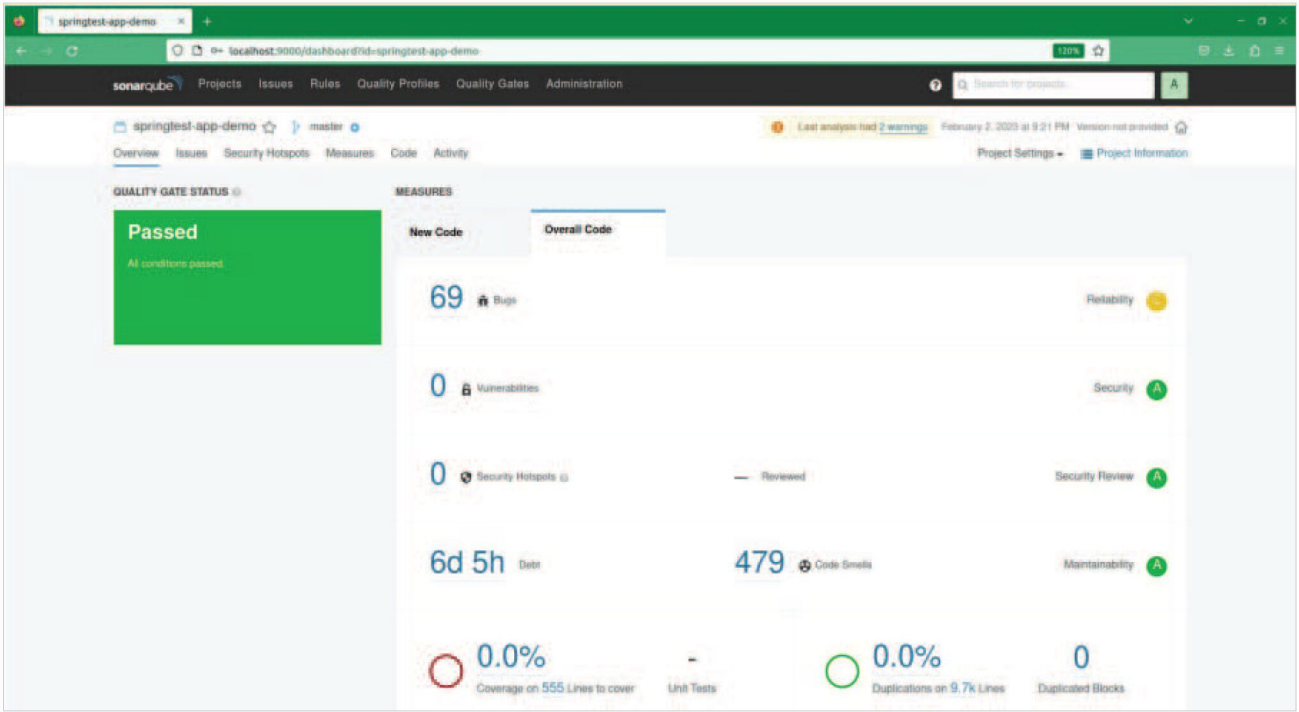



Figure 10: SonarQube overall code


Figure 11: Compliant solution in SonarQube

## Bugs, vulnerabilities and code smells in SonarQube

In SonarQube, the following terms are used to describe different aspects of code quality.

*Bugs:* Functional errors in the code that cause it to behave in incorrect ways.

*Vulnerabilities:* Security risks in the code, such as SQL injection attacks, and cross-site scripting (XSS) attacks.

*Code smells:* Design and architecture issues in the code that cause maintainability problems.

SonarQube provides a summary of the issues.

It also provides detailed information on each individual issue.

We can also find locations in the code and recommended solutions.

By identifying and addressing these issues, SonarQube helps teams to deliver better quality code and improve the overall security and maintainability of their applications in a faster manner. END

### References


- *https://docs.sonarqube.org/latest/*
- *https://docs.sonarqube.org/latest/analyzing-source-code/scanners/sonarscanner/*



**By: Mitesh Soni**

The author has written the books 'Hands-on Azure DevOps, Agile, DevOps and Cloud Computing with Microsoft Azure', 'Hands-on Pipeline as Code with Jenkins', and 'Hands-on Pipeline as YAML with Jenkins'. He has lead with 10 Years of experience in Research & Innovation. He believes in the power of Open Source.

# How to Prevent Cookies from Being Hijacked

Every time you log in to a website, you leave a footprint in the form of cookies. These can be used to gain unauthorised access to the information on your system. Let's take a look at how AES 128 can be used to prevent cookie hijacking.


Image Source: By jcomp on Freepik

Cookies store small blocks of data created by a web server to help the site remember information about your visit. This is helpful for session management during logins and auto-filled form fields, user personalisation (where cookies retain information on user preferences and themes), and for tracking and analysing your web browsing pattern to recommend personalised ads, etc.

You may be wondering how long these cookies retain information. Well, there are different types of cookies. Session cookies store information only till you exit the browser, first-party cookies are stored directly on your computer and help with auto logins and more, secure cookies prevent unauthorised entry, and zombie cookies are present even when they have been deleted or the browser is exited.

The presence of these cookies can lead to unauthorised access to information or services in your computer system. If you look at your normal internet activity, you will find you visit quite a few websites every day. Since we enter sensitive information such as passwords, date of birth, bank account details, etc, when we visit these websites, there are chances this information may fall in the hands of an attacker who is trying to explore the vulnerabilities in the cookies. The attacker may steal the cookie using a fake login or link, put the cookie in the browser, and may fake your identity in the browser. This is known as cookie hijacking or session hijacking because it includes the exploitation of the session key (symmetric key for encrypting data). Common methods used by attackers for

session hijacking include session fixation, session sniffing, cross-site scripting, brute force, etc.

In the project I am working on currently, we leveraged public key cryptosystem (RSA) and device attestation with FIDO2 specifications to ensure extreme security. However, the cookie hijacking vulnerability was still present and had to be dealt with on the server side. So I decided to use AES 128 to check for duplication of the user's IP address. Unlike the RSA algorithm, AES (advanced encryption standard) requires that both the encryptor and decryptor use the same key, which makes symmetric algorithms much faster than the former. AES includes three block ciphers to encrypt and decrypt blocks of messages. Each cipher encrypts and decrypts data in blocks of 128 bits using cryptographic keys of 128 bits. Among the different modes of AES, we used fernet in CBC (cipher block chaining) mode because our project was auto logging off in one hour. So fernet, being smaller in size, was perfect for our use case and its use in the Flask application is given below.

```
from flask import *
from cryptography.fernet import Fernet

key = Fernet.generate_key()
f = Fernet(key)
app = Flask(_name_)
```

In the above code, we are importing Flask and fernet. The class helps with encryption and decryption using the keys generated, which contain bytes or string values. This key is an encoded 32-byte key, with which you may decrypt or encrypt messages.

We may test this with the help of a simple HTML page, like the one shown in Figure 1.

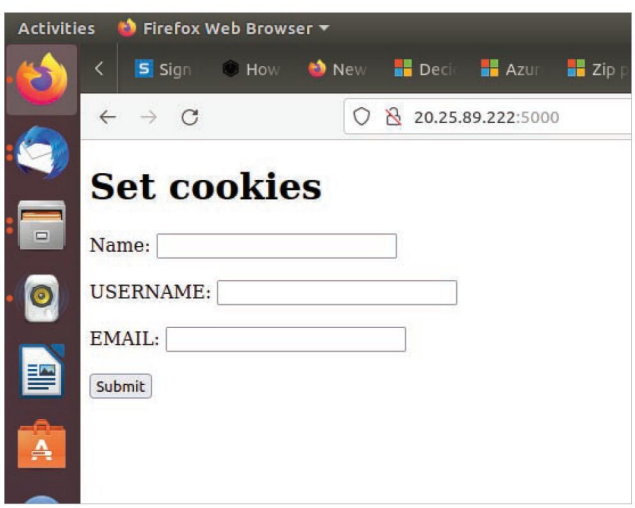



Figure 1: Web page for entering user details

```
@app.route('/login', methods=["GET","POST"])
def login():
        name=request.form['name'].strip()
      uname=request.form['uname'].strip()
        eml=request.form['eml'].strip()
              k=name+'$'+request.remote_
addr+'$'+uname+'$'+eml
      res=f.encrypt(k.encode()).decode()
        resp = make_response(redirect('/dashboard'))
      resp.set_cookie('username', res, max_age=3600)
      return resp
```

In the above code, we are reading from post requests where we get your name, user name and email, and save them after removing spaces.

We can then store the name, IP address, user name and email in the variable *k*, and encrypt it using the generated fernet key.

This gets redirected to the dashboard function. The respective variables containing the encrypted message are set in *cookies* and the maximum age condition is set as 3600.

```
@app.route('/dashboard')
def dashboard():
      username=request.cookies.get('username')
      k=f.decrypt(id.encode()).decode()
      arr=k.split('$')
      name=arr[0]
      ip=arr[1]
      uname=arr[2]
      eml=arr[3]
      if ip==request.remote_addr:
      return render_template('dashboard.html',name=name,uname=
uname,eml=eml)
      else:
      return render_template('error.html',ip=ip,ip1=request.
remote_addr)
app.run()
```

In the dashboard function we decrypt the message from *cookies* and save the data in the respective variables. Next, we check if the IP address provided in the cookie matches the current IP address of the user system. If any case of cookie hijacking is found, you'll see an error page pop up in which the two different IP addresses are displayed and the user gets logged out of the page.

Figure 2 shows the sample page displayed to users after they enter their details. The IP address of the user machine is displayed along with other form details. These details may be misused by gaining unauthorised access to cookie details. In the page shown in Figure 2 we are sending

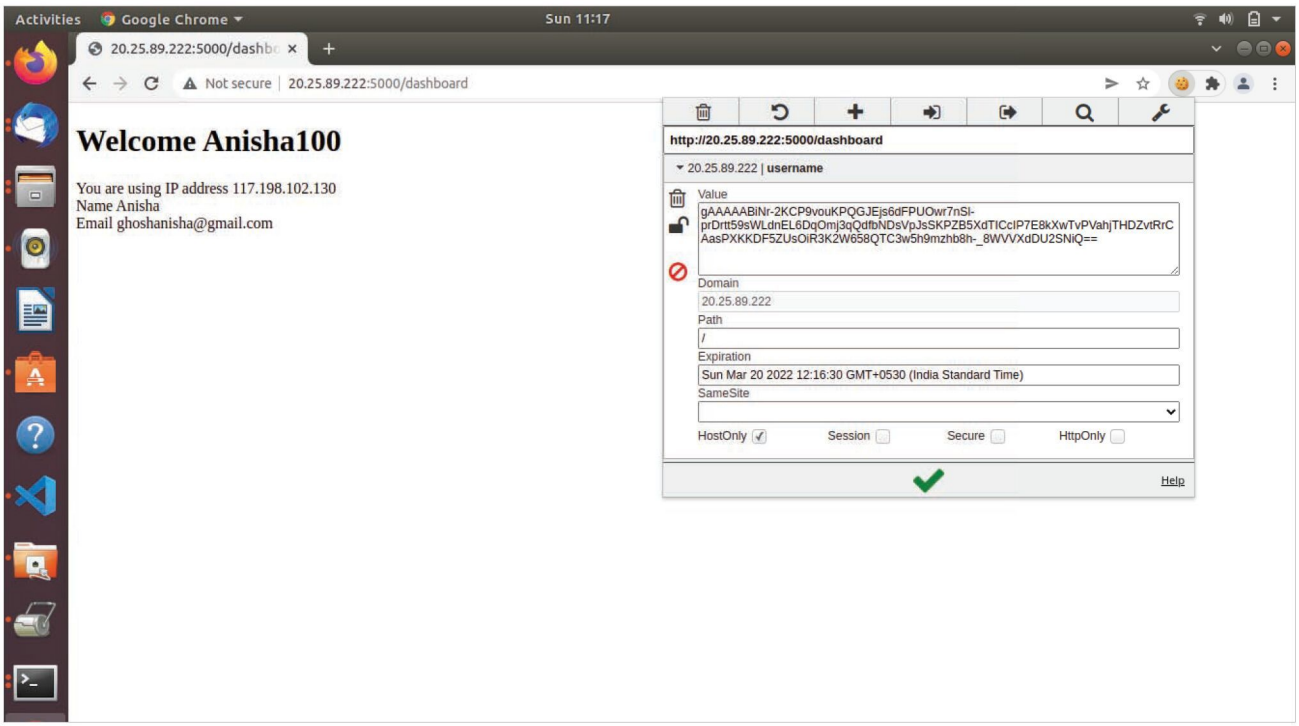



Figure 2: User machine where IP is displayed and cookies are being transferred to another device

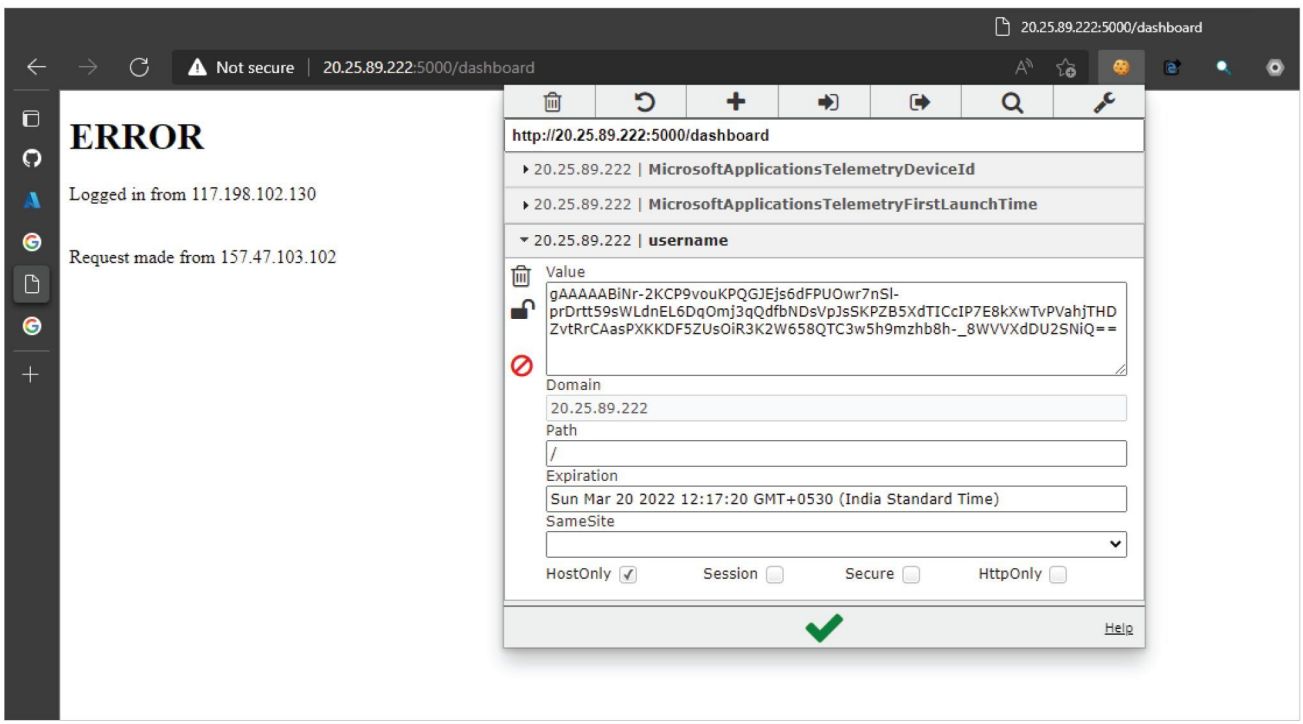


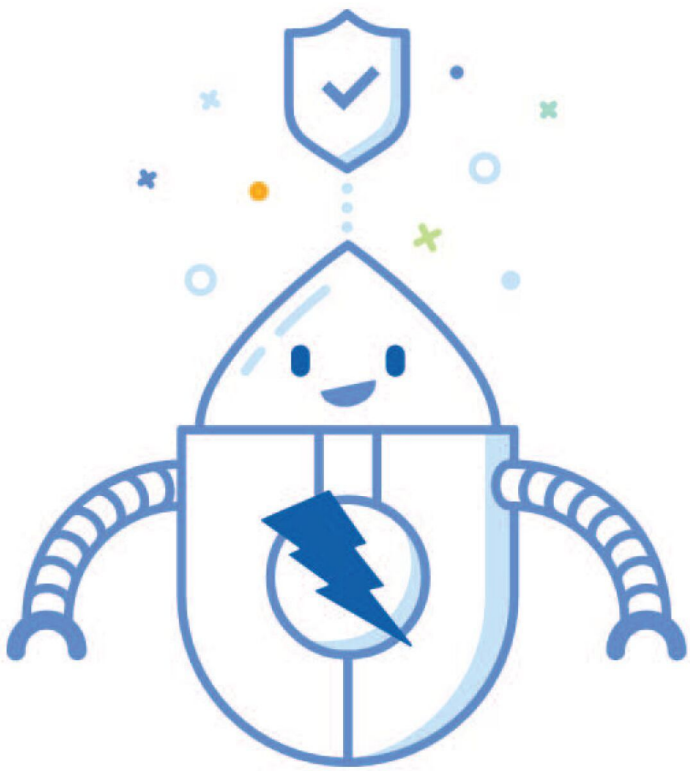

Figure 3: Attacker's machine is denied access

cookie data to another device. In case the attacker gains access and tries to log in by pretending to be the user, the error page is displayed as shown in Figure 3 and cookie hijacking is prevented successfully. END

**By: Anisha Ghosh**

The author is a cyber security researcher and an active contributor to the open source communities and repositories. She is interested in developing novel, scalable and secure systems.

# Dynamic Application Security Testing Using OWASP ZAP



OWASP ZAP is an open source penetration testing tool, which is used to perform dynamic application security testing. Let's learn more about it and find out how to use it.

Dynamic application security testing (DAST) focuses on finding security vulnerabilities in a running application and simulating attacks on it. DAST differs from static application security testing or SAST. The latter focuses on analysing the source code of an application to identify bugs, security vulnerabilities and code smells. The objective of SAST is to identify these issues early in the software development life cycle before they are identified and exploited in the production environment. SAST usually analyses an application's source code, configuration files, infrastructure configuration and build scripts to identify potential bugs and vulnerabilities. We don't need to execute the code to analyse it in SAST.

DAST analyses the behaviour of the deployed application. It finds vulnerabilities and potential risks. DAST tools usually automate the process of simulating attacks such as SQL injection and cross-site scripting (XSS) attacks. DAST is a significant tool for development teams looking to improve the security of applications. We can automate the process of identifying security vulnerabilities in a running application by integrating it in the CI/CD pipeline.

## Overview of OWASP ZAP

In software security testing, we assess and verify a system against security risks and vulnerabilities. System security testing can be categorised in the following way:

- Vulnerability assessment, where scanning and analysing of security risks are performed
- Penetration testing, where simulated malicious attackers attack the system and analyse it
- Runtime testing, where the end user performs security testing
- Code review, where the review and analysis of the system take place to find vulnerabilities

Open Web Application Security Project's (OWASP) Zed Attack Proxy (ZAP) is a flexible, extensible and open source penetration testing tool, also known as a 'man-in-the-middle proxy'. ZAP can intercept and inspect messages sent between a browser and the web application, and perform other operations as well. It is designed to help developers and security professionals identify vulnerabilities in web applications and web services. It can find common web application security issues such as cross-site scripting (XSS), SQL injection, and cross-site request forgery (CSRF). OWASP ZAP also supports plugins that can be used to extend its functionality. Its ease of use, a large user base, user-friendly interface and configurability help to meet the specific needs of projects in business units. Given below is a brief description of this tool.

| | |
|---|---|
| Initial release | 2014 |
| Stable release | 2.12.0 |
| Written in | Java |
| License | Apache License 2.0 |
| Website | *https://owasp.org/www-project-zap/* |
| GitHub repository | *https://github.com/zaproxy/zaproxy* |
| Features | • Intercepting proxy<br>• Active and passive scanners<br>• Traditional and ajax spiders<br>• Brute force scanner<br>• Port scanner<br>• Web sockets |
| Risk categories | • High<br>• Medium<br>• Low<br>• Informational<br>• False positive |
| How is it useful in implementing DevOps practices? | Penetration testing |
| Can we integrate it with Pipeline as Code? | Yes |
| Is a commercial flavour available? | N/A |
| ZAP Docker image scan | Docker stable image can be obtained using *docker pull owasp/zap2docker-stable:*<br><br>1. *ZAP - Baseline scan:* This executes the ZAP spider against the specified target for one minute and then completes the passive scanning.<br><br>zap-baseline.py -t <target> [options]<br><br>1. *ZAP - Full scan:* This executes ZAP spider against the specified target, an optional ajax spider scan and then a full active scan.<br><br>zap-full-scan.py -t <target> [options]<br><br>1. *ZAP - API scan:* This is suitable to perform scans against APIs defined by OpenAPI, SOAP, or GraphQL.<br><br>zap-api-scan.py -t <target> -f <format> [options] |

OWASP ZAP offers several benefits for web application security testing:

- It is an open source tool.
- It has a user-friendly interface that makes it easy to perform security testing on web applications and microservices.
- It covers a range of scanning capabilities such as active scanning, passive scanning, and spidering.
- OWASP ZAP can be integrated into continuous integration and continuous delivery (CI/CD) pipelines.
- It has an active community of users and contributors for continuous improvement and innovation.

OWASP ZAP Desktop is a graphical user interface (GUI) for the OWASP ZAP security testing tool. It provides an easy-to-use and user-friendly interface for executing security tests on web applications and microservices.

The ZAP Desktop provides many of the same features and functionality as the command-line version of OWASP ZAP such as the ability to perform automated and manual security tests. Additionally, it offers visual representation of the test results and an easy-to-use interface.

## Running the web application in a minikube cluster

Some of the prerequisites for installing minikube on an Ubuntu operating system are:

| Containers or virtual machine managers | Architecture |
|---|---|
| • Docker, QEMU, Hyperkit, Hyper-V, KVM, Parallels, Podman, VirtualBox, or VMware Fusion/Workstation | • 2 CPUs or more<br>• 2GB of free memory<br>• 20GB of free disk space |

To install the latest minikube stable release on Ubuntu 22.04.1 LTS, execute the following commands:

```
osfy@ubuntu:~/Desktop$ curl -LO https://storage.googleapis.
com/minikube/releases/latest/minikube_latest_amd64.deb
  % Total    % Received % Xferd  Average Speed   Time    Time
Time  Current
                 Dload  Upload   Total   Spent    Left  Speed
100 26.1M  100 26.1M    0     0  1430k      0  0:00:18
0:00:18 --:--:-- 1231k

osfy@ubuntu:~/Desktop$ sudo dpkg -i minikube_latest_amd64.deb

Selecting previously unselected package minikube.
(Reading database ... 204624 files and directories currently
installed.)
Preparing to unpack minikube_latest_amd64.deb ...
Unpacking minikube (1.28.0-0) ...
Setting up minikube (1.28.0-0) ...
```

Start your minikube cluster using the minikube *start* command. If you face any issues, stop the cluster, delete it and execute the minikube *start* command again. The next step is to verify Java and Maven installation, create a Docker image, and deploy the Docker image using Kubernetes YAML in the minikube cluster.

Next, install Java and Maven in Ubuntu OS. Once installation is completed successfully, verify the Java version using the *java –version* command.

```
osfy@ubuntu:~/Desktop$ java -version

openjdk version "11.0.17" 2022-10-18
OpenJDK Runtime Environment (build 11.0.17+8-post-Ubuntu-
1ubuntu222.04)
OpenJDK 64-Bit Server VM (build 11.0.17+8-post-Ubuntu-
1ubuntu222.04, mixed mode, sharing)
```

Now, verify the Maven version using the *mvn -version* command:

```
osfy@ubuntu:~/Desktop$ mvn -version
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 11.0.17, vendor: Ubuntu, runtime: /usr/lib/jvm/
java-11-openjdk-amd64
Default locale: en_IN, platform encoding: UTF-8
OS name: "linux", version: "5.15.0-53-generic", arch:
"amd64", family: "unix"
```

We are going to use Petclinic as a sample application, which is a Spring Boot application built using Maven or Gradle. In this article we will build a JAR file and deploy it in a minikube cluster. Visit *https://github.com/spring-projects/spring-petclinic* to know more. Copy the GitHub repository URL from the above URL, and use the *git clone* command to clone it locally.

```
osfy@ubuntu:~/Desktop$ git clone https://github.com/spring-
projects/spring-petclinic.git

Cloning into 'spring-petclinic'...
remote: Enumerating objects: 9424, done.
remote: Total 9424 (delta 0), reused 0 (delta 0), pack-reused
9424
Receiving objects: 100% (9424/9424), 7.64 MiB | 1.55 MiB/s,
done.
Resolving deltas: 100% (3567/3567), done.
```

Once the source code is available, go to the root of the project using the *change directory* command as given below, and execute the *./mvnw package* command to create a JAR file. At first execution it will try to download multiple dependencies, and hence will take some time to complete execution.

```
osfy@ubuntu:~/Desktop$ cd spring-petclinic
osfy@ubuntu:~/Desktop/spring-petclinic$ ./mvnw package
Warning: JAVA_HOME environment variable is not set.
[INFO] Scanning for projects...
Downloading from spring-snapshots:
[INFO]
[INFO] ------------< org.springframework.samples:spring-
petclinic >------------
[INFO] Building petclinic 2.7.3
[INFO] --------------------------------[ jar ]---------------
.
.
[INFO] Building jar: /home/osfy/Desktop/spring-petclinic/
target/spring-petclinic-2.7.3.jar
[INFO]
[INFO] --- spring-boot-maven-plugin:2.7.3:repackage
(repackage) @ spring-petclinic ---
[INFO] Replacing main artifact with repackaged archive
[INFO] ------------------------------------------------------
```

```
[INFO] BUILD SUCCESS
```

Observe the logs of Maven command execution where it says *Building jar: /home/osfy/Desktop/spring-petclinic/target/spring-petclinic-2.7.3.jar.*

Now, we have a JAR file ready. The next step is to create a Docker image that contains the JAR file and runtime environment.

So, what is a Dockerfile? The Dockerfile is nothing but the text document that provides build instructions to build the image with your application package.

```
FROM openjdk:11
MAINTAINER osfy
COPY target/spring-petclinic-2.7.3.jar app.jar
ENTRYPOINT ["java","-jar","/app.jar"]
```

Place the Dockerfile in the project root directory. Now, it is time to build a Docker image.

For testing locally with minikube, we should point the local Docker daemon to the minikube internal Docker registry, and build the image to install it in the minikube cluster: *eval $(minikube docker-env).*

Next, execute *sudo docker image build -t petclinic:latest*, which is the command from the project root directory.

We will use Docker images available in the local environment and not from the public or private Docker registry; so set the imagePullPolicy value to *Never* in the YAML file. Given below is the YAML file for deployment and service, to deploy a sample application in the minikube cluster.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: petclinic
  name: petclinic
spec:
  replicas: 1
  selector:
    matchLabels:
      app: petclinic
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: petclinic
    spec:
      containers:
      - image: petclinic:latest
        name: petclinic
        resources: {}
        imagePullPolicy: Never
status: {}
---
apiVersion: v1
kind: Service
metadata:
  name: petclinic
spec:
  type: LoadBalancer
  selector:
    app: petclinic
  ports:
    - protocol: TCP
      port: 8080
      targetPort: 8080
```

Use the *kubectl apply* command to create service and deployment available in the YAML file.

```
osfy@ubuntu:~/Desktop$ kubectl apply -f petclinic-
deployment.yaml

deployment.apps/petclinic created
service/petclinic created
```

minikube tunnel runs as a process that helps to create a network route on the host to the service CIDR of the cluster, using the cluster's IP address as a gateway. The *tunnel* command exposes the external IP directly to any program running on the host operating system. Open a new terminal window and execute the *minikube tunnel* command.

```
osfy@ubuntu:~/Desktop$ minikube tunnel
[sudo] password for osfy:
Status:
     machine: minikube
     pid: 94645
     route: 10.96.0.0/12 -> 192.168.49.2
     minikube: Running
     services: [petclinic]
    errors:
       minikube: no errors
       router: no errors
       loadbalancer emulator: no errors

Let's verify services:

osfy@ubuntu:~/Desktop$ kubectl get services
```

```
NAME         TYPE           CLUSTER-IP     EXTERNAL-IP
PORT(S)          AGE
kubernetes   ClusterIP      10.96.0.1      <none>
443/TCP          12m
petclinic    LoadBalancer   10.98.225.67   10.98.225.67
8080:30200/TCP   69s
```

Visit the localhost:8080 in the browser and our sample application is ready.

## Installing ZAP

OWASP ZAP can be installed on different kinds of operating systems such as Windows, macOS, and Linux. The exact steps for installation depend on the operating system.

Click on *Linux Installer* and it will download ZAP_2_12_0_unix.sh.

Change the access permission of ZAP_2_12_0_unix.sh using the *chmod* command and run it. You need to be a root user to run it (as shown in Figure 1).

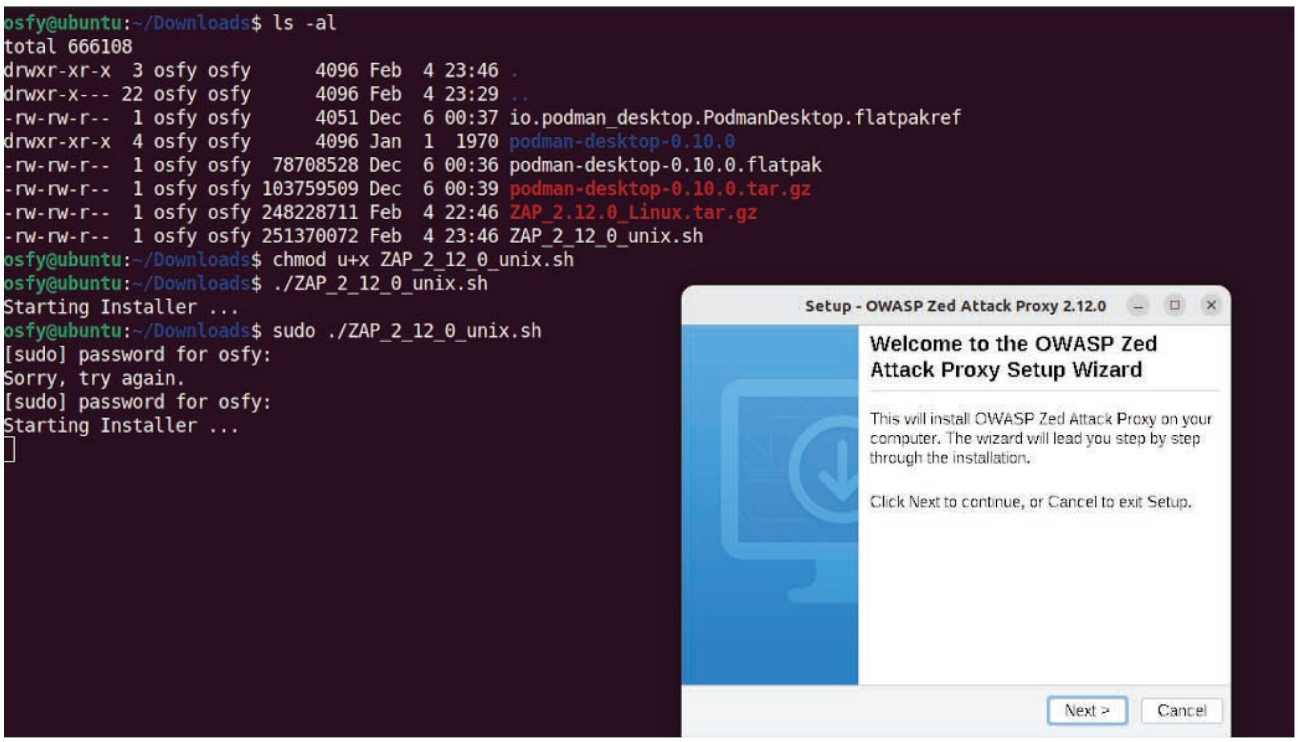



Figure 1: Installing ZAP on Linux

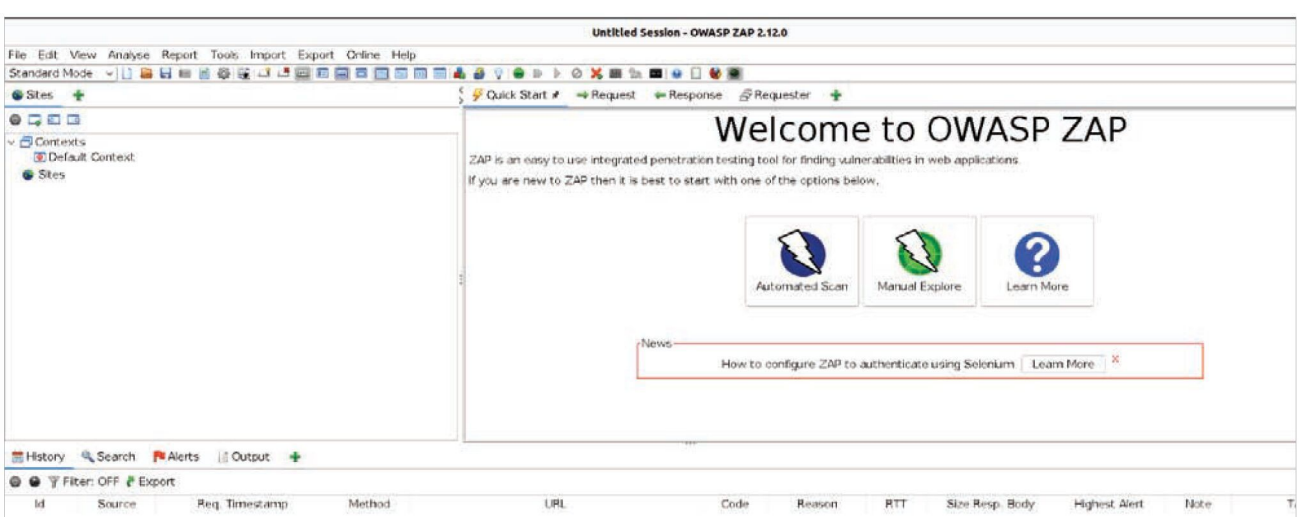



Figure 2: Application in action

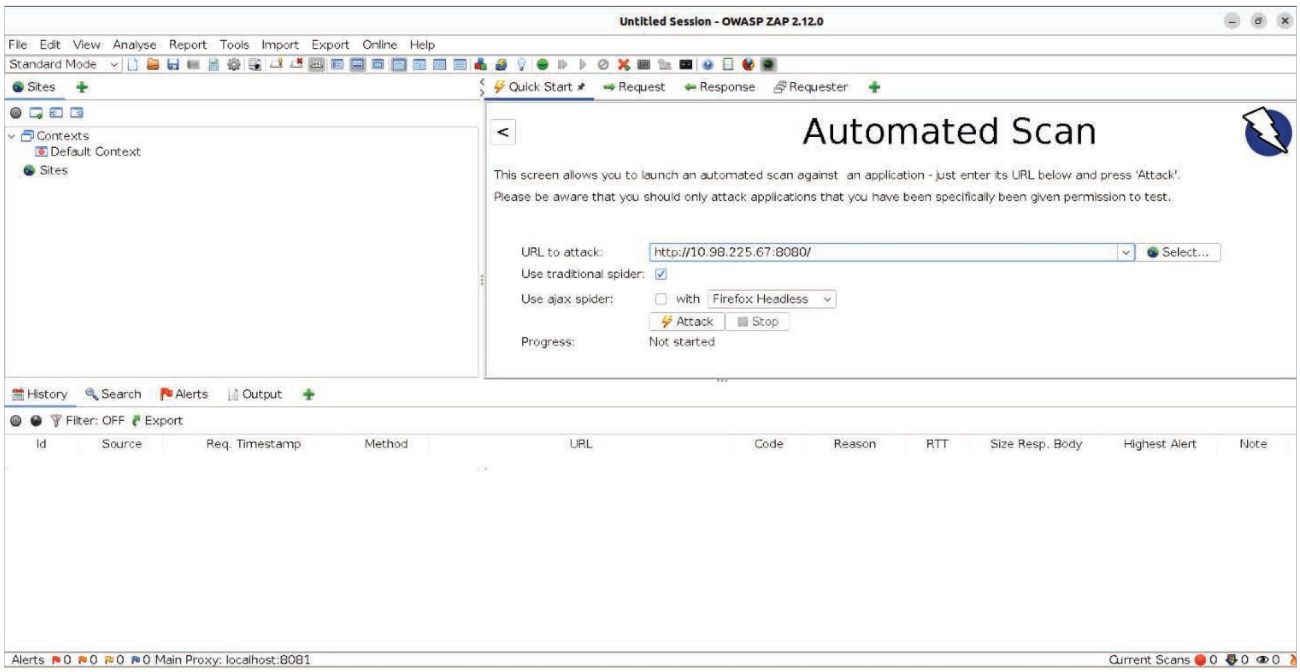



Figure 3: Starting the attack

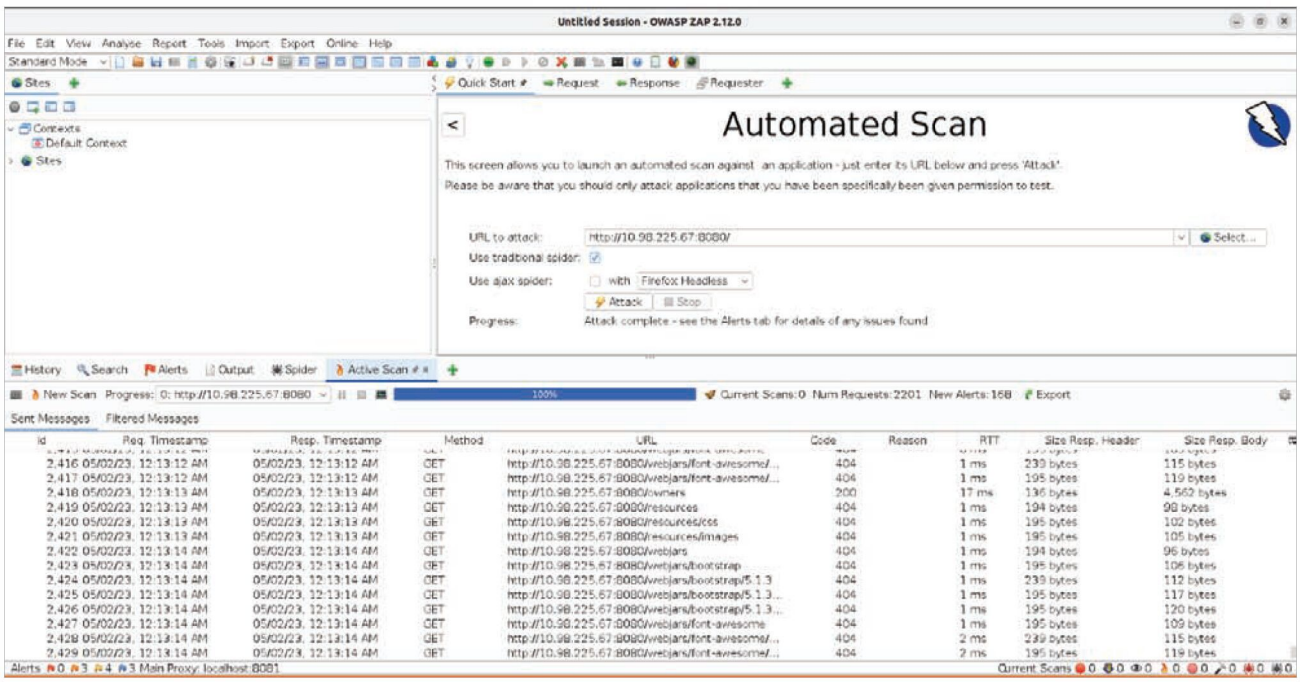



Figure 4: Generating a report

Follow the installation steps and finish the installation.

Run the application once the installation is completed and then click on 'Automated Scan', as shown in Figure 2.

Give the application URL that we have deployed in the minikube cluster and click on 'Attack', as shown in Figure 3.

Wait until 'Active Scan' is completed. Now click on 'Report Menu' and select 'Generate Report'. Select the appropriate 'Template' (Figure 4).

You can even generate the PDF of the scanning report.

OWASP ZAP can also be run using Docker. In fact, Docker provides a convenient way to run OWASP ZAP in a consistent environment, regardless of the underlying operating system. END

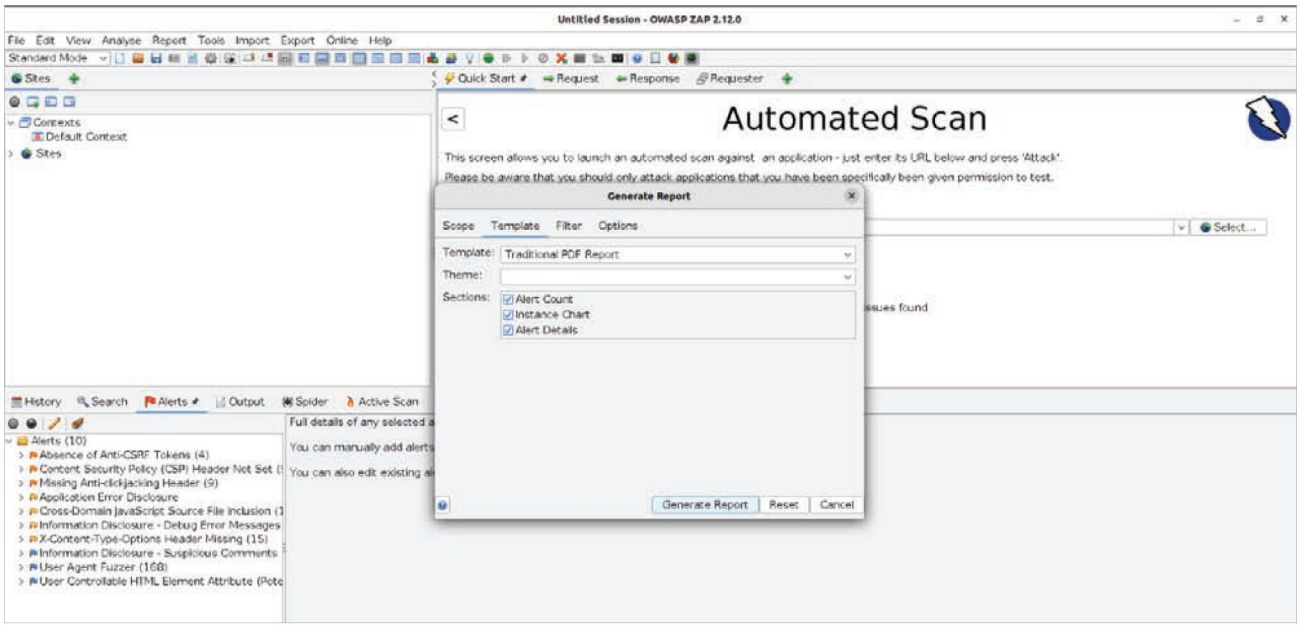



Figure 5: Generating PDF

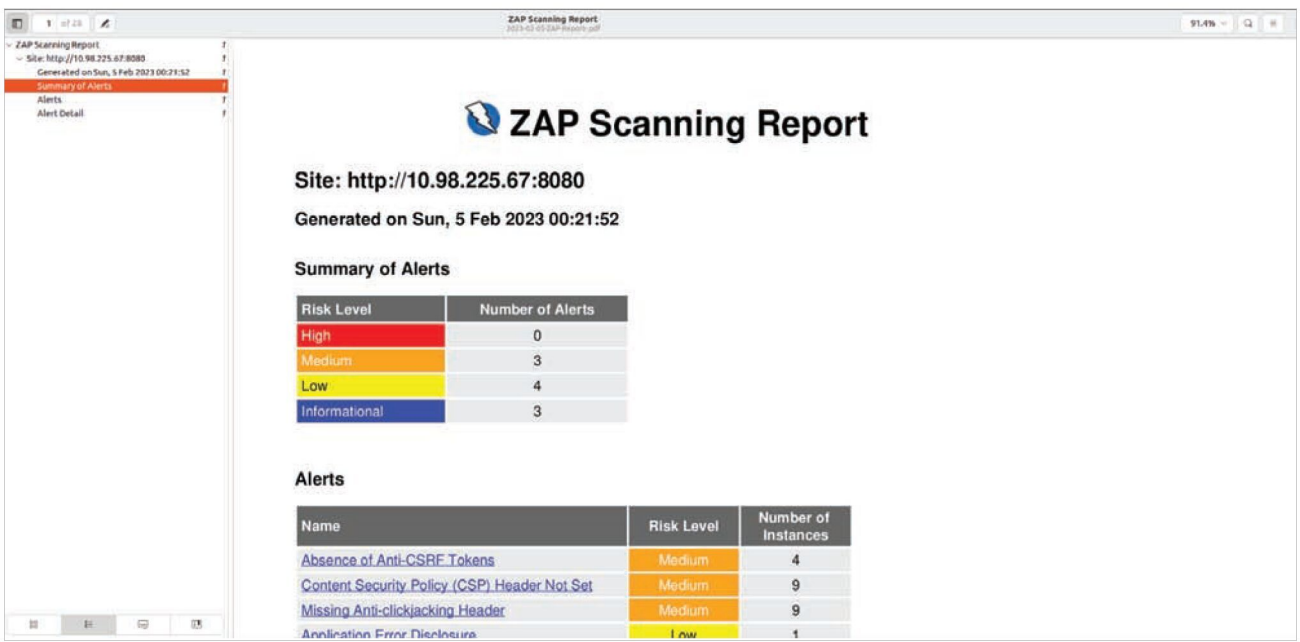



Figure 6: Complete scanning report

## References


- Minikube Documentation, *https://minikube.sigs.k8s.io/docs/start/*
- Docker: Accelerated, Containerized Application Development, *https://www.docker.com*
- Docker Documentation, *https://docs.docker.com*
- A sample Spring-based application, *https://github.com/spring-projects/spring-petclinic*
- Deployments, *https://kubernetes.io/docs/concepts/workloads/controllers/deployment/*
- Service, *https://kubernetes.io/docs/concepts/services-networking/service/*



**By: Mitesh Soni**

The author has written the books 'Hands-on Azure DevOps, Agile, DevOps and Cloud Computing with Microsoft Azure', 'Hands-on Pipeline as Code with Jenkins', and 'Hands-on Pipeline as YAML with Jenkins'. He has lead with 10 Years of experience in Research & Innovation. He believes in the power of Open Source.

# AI: An Introduction to Natural Language Processing

In the last article in this series on AI and machine learning, we developed further insights on the workings of neural networks. Subsequently, we enhanced our understanding of unsupervised learning through the use of scikit-learn. Finally, we embarked on learning PyTorch, a machine learning framework built on the Torch library. In this ninth article in the series on AI, we will focus on a topic that has not yet been covered -- natural language processing (NLP). For this purpose, we will utilise both PyTorch and the NLTK (natural language toolkit) libraries. We will also discuss PySpark, an interface for Apache Spark in Python. However, as always, we will start by solidifying our theoretical grasp of AI and machine learning.

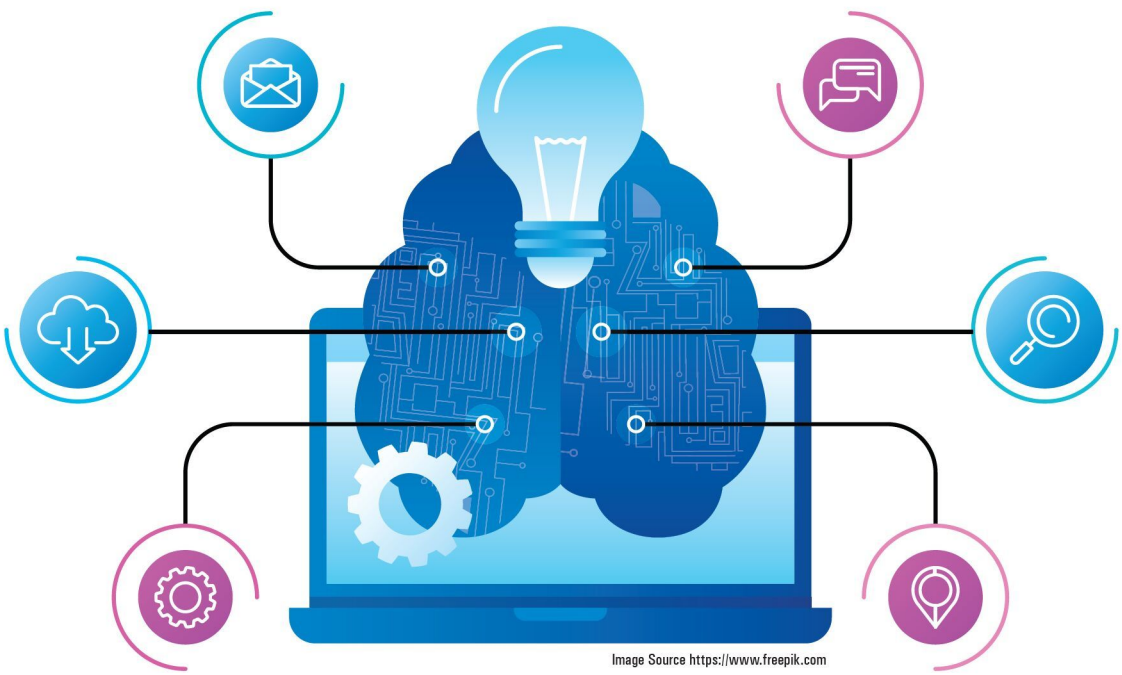


Image Source https://www.freepik.com

Before we delve deeper into our discussion of neural networks, let us take a moment to consider the current state of affairs. If you examine the recent trends in the Indian engineering education sector, you will see a growing interest among young students to study computer science engineering. Although some tech companies have started layoffs, I believe the fascination with computer science engineering among the younger generation will not decline in the near future. Of course, I may be wrong, and only time can tell. Now, what about the areas of interest among students who have recently begun studying computer science engineering? Many students today are interested only in fields like AI, machine learning, and data science. This choice is understandable, given the emergence of AI based tools like ChatGPT. The abundance of literature in this field can make it difficult to keep up with the new terminology that is constantly being introduced. I recently participated in evaluating several undergraduate level academic projects based on AI and machine learning. The amount of new jargon presented by the project groups was both fascinating and concerning. Given that many of our bright young students plan to work in the fields of AI and machine learning in the near future,

it is essential for us to have a good understanding of the popular jargon in these fields. That is why I believe our discussions on AI, machine learning, data science, neural networks, and related topics are well justified.

Recall that in the last few articles in this series, we thoroughly covered neural networks. We talked about using TensorFlow and Keras to build and test models based on neural networks. However, it's important to note that our approach to neural networks was largely practical in nature. Remember that in the previous two articles in this series, we created and examined a neural network model capable of classifying images of handwritten digits. Now, I believe it is the right time to delve into the various types of neural networks that can be employed in such models.

The origin of neural networks can be traced back to the early 1940s, prior to the development of digital computers itself. In 1943, Warren McCulloch and Walter Pitts introduced perceptron (also called McCulloch-Pitts neuron), a mathematical representation of a simple neuron capable of processing and transmitting information like the neurons in the human brain. Despite the groundbreaking proposal, neural networks didn't immediately become the dominant technology in the field of AI. Instead, other techniques such as support vector machines and linear regression were favoured in the development of AI and machine learning applications. It wasn't until the 1990s that neural networks saw a resurgence in popularity. In the next section, we will explore some of the commonly used neural networks and list the corresponding Python libraries for implementing them.

First, let us discuss feedforward neural networks (FNN). FNNs are the simplest of all the neural networks. In an FNN, information flows only in one direction — from the input layer to the output layer. These networks are used for a wide range of tasks, including image classification and regression analysis. A single-layer perceptron is one of the simplest FNNs, consisting of just an input layer and an output layer. They are often used for supervised learning of binary classifiers. There are also multilayer perceptrons which are far more powerful than single-layer perceptrons. Unlike single-layer perceptrons, multilayer perceptrons have one or more hidden layers in addition to an input layer and an output layer. Multilayer perceptrons are often used for regression analysis. TensorFlow/Keras and PyTorch can be used to implement FNNs in Python. Consider the Python script named *mlp.py* shown below which uses a library called *torchvision* — part of the PyTorch framework. If this library is not part of the version of PyTorch installed in your system, use Anaconda Navigator to install it. The Python script *mlp.py* generates a multi-layer perceptron (MLP). Figure 1 shows the output of the Python script named *mlp.py*.

```
import torch
import torchvision.ops as ops

mlp = ops.MLP(in_channels=5, hidden_
channels=[5, 5])
inp = torch.randn(1, 5)
print(inp)
output = mlp(inp)
print(output)
```

Now, let us discuss convolutional neural networks (CNN). A CNN is a type of FNN. CNNs are specifically designed to process image data. They are often used for image classification, object detection, image segmentation, etc. TensorFlow/Keras and PyTorch can be used to implement CNNs in Python. CNNs are made of a number of convolutional layers. Recall that our model for classifying images of handwritten digits used convolutional layers. Consider the Python script named *cnn.py* shown below. This Python script generates a 2D convolution layer using PyTorch. Figure 2 shows the output of the Python script named *cnn.py*.

```
import torch
import torch.nn as nn
model = nn.Conv2d(4, 1, 1)
inp = torch.randn(4, 1, 1)
print(inp)
output = model(inp)
print(output)
```

Now, let's discuss recurrent neural networks (RNN) and long short-term memory networks (LSTM). RNNs have loops that enable information to be passed from one step to the next. LSTM, on the other hand, is a more sophisticated type of RNN

```
# python mlp.py
tensor([[0.0796, 1.3288, 0.5934, 1.1947, 0.8144]])
tensor([[ 0.1073, -0.1165, -0.0880,  0.8064,  0.2044]],
       grad_fn=<AddmmBackward0>)
```

Figure 1: The working of a multilayer perceptron

```
# python cnn.py
tensor([[[ 0.2679]],

        [[ 0.0545]],

        [[ 0.9483]],

        [[-1.4280]]])
tensor([[[-0.6475]]], grad_fn=<SqueezeBackward1>)
```

Figure 2: The working of a CNN

```
# python torch1.py
['be', 'the', 'change', 'you', 'wish', 'to', 'see', 'in', 'the', 'world']
['be', 'the', 'change', 'you', 'wish', 'to', 'see', 'in', 'the', 'world', 'be
 the', 'the change', 'change you', 'you wish', 'wish to', 'to see', 'see in',
 'in the', 'the world']
```

Figure 3: PyTorch for NLP

that uses specialised memory cells and gates to selectively remember or forget information over time. Thus, RNN and LSTM can be referred to as feedback neural networks. Both are neural networks specifically designed to process sequential data such as natural language text. LSTMs can be implemented in Python using TensorFlow/Keras or PyTorch. The class *tf.keras.layers.LSTM( )* generates LSTM layers using TensorFlow/Keras and the class *torch.nn.LSTM( )* generates LSTM layers using PyTorch. Notice that FNNs, CNNs, RNNs, and LSTMs are used to implement supervised learning models.

Let us now turn our attention to generative adversarial networks (GAN), which are a specific type of neural network used for generative modelling. A GAN is composed of two separate neural networks -- a generator and a discriminator. The generator is trained to create new data that resembles the training data, while the discriminator is trained to distinguish between the generated data and real data. It is worth noting that GANs are distinct from the other neural networks we have discussed so far in that they are used for unsupervised learning. GANs can be implemented in Python using TensorFlow/Keras or PyTorch. GANs typically rely on a variety of different types of layers, including dense layers, convolutional layers, convolutional transpose layers, activation layers, and dropout layers. For example, in TensorFlow/Keras, the *tf.keras.layers* module provides a range of pre-built layers that can be used for implementing GANs, while PyTorch offers similar functionality through its *torch.nn* module.

Let us now turn our attention to autoencoders. They are another type of neural network used for unsupervised learning. An autoencoder is made up of two parts, an encoder and a decoder. Autoencoders are trained to reconstruct the input data. They are often used for tasks such as anomaly detection and dimensionality reduction. Python offers several ways to implement autoencoders, which includes usage of TensorFlow/Keras or PyTorch. The implementation of autoencoders can be done by using the *tf.keras.module* module in TensorFlow/Keras or by using the *torch.nn* module in PyTorch.

Besides the neural networks that we have discussed, there are numerous other notable neural networks to be considered. Some of these include the Hopfield network, which was introduced in 1982, and is a type of recurrent neural network that stores and retrieves patterns; Boltzmann machines (BM), which are stochastic neural networks that represent complex probability distributions; restricted Boltzmann machines (RBM) introduced in 1986, which are Boltzmann machines with a restricted architecture that facilitates efficient training; and deep belief networks (DBNs), which are generative models with multiple layers of RBMs. Let us now shift our focus to the more practical aspects of developing applications based on AI and machine learning.

## NLP using PyTorch

First, let us discuss natural language processing or NLP, which plays a critical role in automation. Notice that NLP faces significant challenges due to the sheer number of languages spoken globally — eight languages are spoken by more than 100 million people each, and 27 languages are spoken by at least 50 million people each (source: Wikipedia). These languages differ in their grammatical structures, idiomatic expressions, and linguistic features, making analysis and modelling difficult. Additionally, in our globalised world, text often contains words from multiple languages, some of which may even be from relatively obscure languages. Processing the vast amount of data involved in NLP presents another significant challenge, along with the inherent ambiguity of natural languages. Despite these difficulties, we have access to excellent tools for NLP such as Python powered with PyTorch. Now, let us briefly discuss the use of PyTorch for NLP.

Recall that we have already seen examples of how PyTorch can be used for building neural networks, which are commonly employed in NLP. In this section, we will examine some of the key PyTorch modules and classes that are particularly useful for NLP. Specifically, the *torch.nn* module offers a range of classes for building neural networks such as CNN, RNN, etc, as well as activation functions like ReLU, Sigmoid, etc. The *torch.optim* module provides various optimisation algorithms such as stochastic gradient descent (SGD), RMSprop, Adam, etc, which can be used to train neural networks. Recall that we used the Adam optimisation algorithm in our neural network model to classify images of handwritten digits. However, keep in mind that our model was implemented using Keras. The *torch.utils.data* module provides a set of tools for working

with data sets and loading data. The main classes provided by *torch.utils.data* are *Dataset* and *DataLoader*, which provide interfaces for accessing data and loading samples from a data set, respectively.

PyTorch provides not only the above mentioned standard features but also a dedicated library for NLP called TorchText. TorchText offers a number of data processing functions and several NLP data sets. It is composed of five main classes: *data*, *datasets*, *nn*, *utils*, and *vocab*. To illustrate the working, let us take a look at a simple example that demonstrates the use of TorchText (imported as *torchtext*) for text processing. Consider the Python script named *torch1.py* shown below (line numbers are added for ease of explanation).

```
1. import torchtext
2. from torchtext.data import get_
tokenizer
3. from torchtext.data.utils import
ngrams_iterator
4. tokenizer = get_tokenizer("basic_
english")
5. tokens = tokenizer("Be the change
you wish to see in the world")
6. print(tokens)
7. print(list(ngrams_iterator(tokens,
2)))
```

The Python script shown above finds the tokens in a string and prints all the bigrams formed from the tokens identified. But first, we need to understand what an n-gram is. An n-gram is a continuous sequence of *n* tokens from a given sample of text or speech. Bigrams are a special case of the n-gram, where n is 2. But why are bigrams or n-grams critical in NLP? Well, often a lot more information can be obtained from a text or speech by considering n-grams over individual words. For example, if a word is preceded by the definite article 'the', typically this indicates that the word is being used in a specific sense rather than in a general sense. For example, consider the string '*the book*'; it refers to a specific book that has been previously mentioned whereas the word '*a book*' refers to any book without specifying a particular one.

Now, let us go through the code line by line. In our script *torch1.py*, we are considering words in the string as tokens. Line 1 imports the library *torchtext*. Lines 2 and 3 import the methods *get_tokenizer( )* and *ngrams_iterator( )*. Line 4 defines a tokeniser which splits the string into tokens after normalisation. Line 5 applies the tokeniser on the string '*Be the change you wish to see in the world*'. Line 6 prints the tokens generated. Finally, Line 7 prints all the bigrams formed from the tokens. Figure 3 shows the output of the Python script *torch1.py*. If you replace Line 7 with the line of code, '*print(list(ngrams_iterator(tokens, 3)))*', then all the trigrams formed from the tokens will be printed. Trigrams are another special case of the n-gram, where n is 3. PyTorch is a general machine learning framework with a lot of features for NLP. However, there are also libraries designed specifically for NLP. Next, we discuss one such library called natural language toolkit (NLTK).

## NLP using NLTK

NLTK is a symbolic and statistical NLP Python library. It is free and open source software licensed under Apache License version 2.0. NLTK can be used for text processing, tokenisation, part-of-speech tagging, sentiment analysis, etc. As usual, the installation of NLTK can be done easily with the help of Anaconda Navigator. Now, let us see how a Python script similar to *torch1.py* can be written using NLTK. Consider the program *nltk1.py* shown below, which finds the tokens in a string and prints all the bigrams formed from the tokens identified (line numbers are added for ease of explanation).

```
1. import nltk
2. from nltk.tokenize import word_
tokenize
3. tokens = nltk.word_tokenize("Be the
change you wish to see in the world")
4. print(tokens)
5. print(list(nltk.ngrams(tokens, 2)))
```

Now let us go through the code line by line. Line 1 imports the *nltk* library. Line 2 imports the method *word_tokenize( )*. Line 3 applies the function *word_tokenize( )* on the string '*Be the change you wish to see in the world*'. Line 4 prints the generated tokens, and finally, Line 5 prints all the bigrams formed from the tokens. Figure 4 shows the output of the Python script *nltk1.py*. If you replace line 5 with the line of code '*print(list(nltk.ngrams(tokens, 3)))*', then all the trigrams formed from the tokens will be printed. Notice that NLTK also provides a function called *bigrams( )* which will generate all the bigrams from a set of tokens.

```
# python nltk1.py
['Be', 'the', 'change', 'you', 'wish', 'to', 'see', 'in', 'the', 'world']
[('Be', 'the'), ('the', 'change'), ('change', 'you'), ('you', 'wish'), ('wish',
'to'), ('to', 'see'), ('see', 'in'), ('in', 'the'), ('the', 'world')]
```

Figure 4: Finding bigrams with NLTK

Now, let us familiarise ourselves with a Python script that prints the parse tree of a string. First, let us try to understand what a parse tree is. A parse tree represents the syntactic structure of a sentence in a graphical format, where each node in the tree represents a part of the sentence. The tree is constructed by breaking down the sentence into its parts and grouping them into phrases based on their grammatical relationships. Parse trees are important in NLP because they help to understand the grammatical structure of a sentence. Now, consider the Python script *nltk2.py* shown below, which generates a parse tree for a given string (line numbers are added for ease of explanation).

```
1. from nltk.corpus.reader.bracket_
parse import BracketParseCorpusReader
2. corpus_root = '/media/deepu/Data'
3. corpus =
BracketParseCorpusReader(corpus_root,
r'.*\.mrg')
4. tree = corpus.parsed_sents('test.
mrg')[0]
5. tree.draw()
```

Before trying to understand how the code works, we need to discuss treebanks. According to Wikipedia, "In linguistics, a treebank is a parsed text corpus that annotates syntactic or semantic sentence structure." Not a very simple explanation, I suppose. Well, think of treebanks as data sets that tag different words as one among the different parts of speech in English. Of course, this is an oversimplification. In English, the eight parts of speech are noun, verb, adjective, adverb, pronoun, preposition, conjunction, and interjection. If you are not familiar with these concepts, I suggest you pause and read at least the Wikipedia article titled 'Parts of speech'. If you plan to work extensively on NLP, I suggest that you receive proper training from a linguistics expert. One of the most popular treebanks available for use is the Penn treebank (PTB), a data set maintained by the University of Pennsylvania. PTB contains over four million annotated words. To generate parse trees, sample files from PTB can be accessed using the method '*treebank.parsed_sents( )*' provided by the module *nltk.corpus*. However, the sample files available in PTB are relatively large and difficult to understand. Therefore, I have manually created a simple file in PTB format (a file with extension *.mrg*). The sample file, called *test.mrg*, shown below, contains the PTB format for the sentence 'I am Deepu'.

```
(S (NP (PRP I)) (VP (VBP am) (NP (NNP
Deepu))))
```

The file above represents a simple tree structure that can display the given sentence as a hierarchical structure of phrases. First, the subject 'I' is shown as a noun phrase (NP) and the predicate 'am Deepu' is shown as a verb phrase (VP). Later, the next level in the hierarchy shows the word 'I' as a personal pronoun (PRP), the word 'am' as a verb in the base form (VBP) and the word 'Deepu' as a noun phrase (NP). Finally, in the last level, the word 'Deepu' is shown as a proper noun (NNP). Now, let us try to understand the Python script *nltk2.py* line by line. Line 1 imports the *BracketParseCorpusReader( )* method which is used to read files in PTB format. Line 2 stores the path to the directory containing the test file in PTB format. Line 3 uses the *BracketParseCorpusReader( )* method to define a reader capable of reading and processing files in formats like PTB. Line 4 reads the test file named *test.mrg* which is stored in the PTB format. Finally, Line 5 draws the parse tree for the string 'I am Deepu' from the file *test.mrg*. Figure 5 shows the parse tree generated by the Python script *nltk2.py*.

NLTK offers many other features for NLP. Remember that we often have to download data sets and parsers to work with NLTK. An example of a parser that can be downloaded and used while working with NLTK is the Stanford parser. A detailed discussion of parsers and context-free grammars (CFG), which are used to build parsers, is beyond the scope of our discussion. However, for extremely eager students, I suggest the popular textbook 'Introduction to Automata Theory, Languages, and Computation' by John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. But keep in mind that 'Theory of Computation' is a relatively difficult area in computer science to learn. Now, let us shift our focus to data science.

## Introduction to PySpark

If you recall the Venn diagram from the first article in this series, you will know that data science intersects with AI and machine learning. Data science also includes sub-fields like Big Data analytics, which usually doesn't involve AI or machine learning. This section focuses on introducing the development of NLP-based data science applications using PySpark, an interface for Apache Spark in Python. Apache Spark is an open source analytics engine for large scale data processing. It is written in Scala and licensed under Apache License version 2.0.

Let us now talk about the significance of PySpark, which is powered by machine learning. It has three main modules: *pyspark.sql* for structured data and DataFrames, *pyspark.ml* for high-level API machine learning models, and *pyspark.mllib* for low-level API machine learning models. While Apache Hadoop is a widely-used framework for Big Data analytics, it lacks AI and machine learning capabilities. However, PySpark

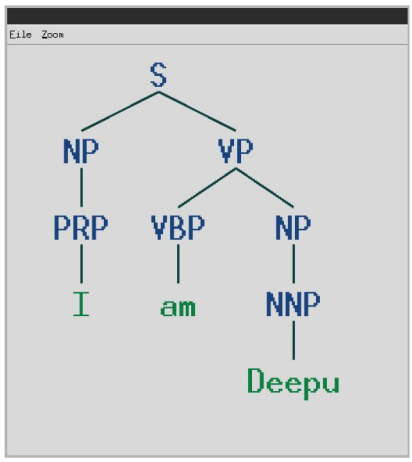



Figure 5: An example for a parse tree

can be combined with Hadoop to provide it with the power of AI and machine learning. PySpark offers distributed computing and APIs for Big Data processing, while Hadoop provides storage, scalability, and fault-tolerance for handling large data sets. This combination with Hadoop makes PySpark an important tool in data science and Big Data analytics.

Now, let us install PySpark in our system. As always, Anaconda Navigator can be used to install PySpark. However, if you execute the command '*pip3 install pyspark[sql, ml, mllib]*' in the terminal, all three important PySpark modules will be installed instantly. You are free to use either of these methods to install PySpark. After installation, execute the command '*pyspark --master local[*]*' in the terminal to launch a PySpark interactive shell with the local machine as the master, while using all available cores for parallel processing. Figure 6 shows the PySpark interactive shell.

Let us now consider the simple Python script named *spark.py* shown below, which uses PySpark to read a CSV file from our system and display the content on the terminal (line numbers are added for ease of explanation).

```
1. from pyspark.sql import
SparkSession
2. spark = SparkSession.builder.
appName("ReadCSV").getOrCreate()
3. df = spark.read.csv("cricket.csv",
header=True, inferSchema=True)
4. df.show(5)
```

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 3.3.1
      /_/

Using Python version 3.9.12 (main, Apr  5 2022 06:56:58)
Spark context Web UI available at http://10.7.45.41:4040
Spark context available as 'sc' (master = local[*], app id = local-1676324644625).
SparkSession available as 'spark'.
>>>
```

Figure 6: The PySpark interactive shell

```
+----------+---------+
|      Name|Centuries|
+----------+---------+
|    Sachin|       49|
|     Kohli|       43|
|   Ponting|       30|
|     Rohit|       29|
|Jayasuriya|       28|
+----------+---------+
only showing top 5 rows
```

Figure 7: Output of the Python script *spark.py*

First, let us try to understand the working of the Python script *spark.py*. Line 1 imports the *SparkSession* class from the *pyspark.sql* module in the PySpark library. The *SparkSession* class provides methods for manipulating data with PySpark, and is used to create data frames and execute SQL queries in a distributed computing environment. Line 2 creates a *SparkSession* object named *spark*. The *getOrCreate( )* method is used to either get the existing *SparkSession* or create a new one if it does not exist. Line 3 reads a CSV file named *cricket.csv*. Finally, Line 4 prints the first five rows of the CSV file *cricket.csv*. Figure 7 shows the output of the Python script *spark.py*.

Now let us address a crucial question: We have previously performed the same tasks using Pandas quite some time ago. In that case, what is the need for PySpark? While Pandas is a great tool for handling and analysing data from small to medium-sized data sets, PySpark is a scalable distributed computing platform designed to manage large data volumes. If your data set is large, PySpark is the better choice. Pandas is only suitable for small to medium-sized data sets that do not require a distributed system.

Now, it is time to wind up this article. Here we first delved deep into the theory of neural networks and then turned our attention to NLP. We utilised PyTorch and NLTK to gain an understanding of NLP. Afterwards, we introduced PySpark and thus marked our first visit into the fields of data science and Big Data analytics in this series. In the upcoming article in this series, we will focus on computer vision using AI and machine learning, and reacquaint ourselves with OpenCV, a library for real-time computer vision. END

**By: Deepu Benson**

The author is currently working as assistant professor in the Amal Jyothi College of Engineering, Kanjirappally, Kerala. He is a free software enthusiast and his area of interest is theoretical computer science.

# How to Convert ChatGPT into an Advanced Voice Assistant

ChatGPT needs no introduction. You ask it any question and it replies in a flash. But the answer comes in the form of text. What if you could talk with it, just like you do with any voice assistant like Siri?

It's no secret that ChatGPT has revolutionised the world of AI. Unlike other AI bots, it is able to understand the context of a conversation and respond, and it makes you feel like you are chatting with a human and not a machine.

But as it is still a kind of chatbot, you need to type a question and you get the answer in the form of text. That's not as exciting as talking to a bot.

This thought gave me the idea of programming ChatGPT so that it could be used as a voice assistant called VoiceGPT. I began by using natural language processing (NLP) to recognise the voice, and then transferring the recognised voice to the ChatGPT engine as a query using the API. After getting an intelligent reply from ChatGPT, I again used NLP to convert it into a human voice.

I needed a good NLP tool for this and OpenAI itself provides one, i.e., Whisper. But due to limited time and space, I ended up using Google Natural Language API.

## A step-by-step guide to making VoiceGPT

We need to begin by combining the NLP for the ttX service with ChatGPT. For this we need a machine to run the open API, transfer the query gathered from NLP, and reprocess the answer given by ChatGPT into a human voice using NLP.

You can use any laptop, but I chose the Raspberry Pi to run all this. For capturing the voice for recognition, I attached the voice bonnet; a USB microphone can also be used with Raspberry Pi. However, if you are using a laptop to run the VoiceGPT code, there is no need for a USB microphone; you can use the laptop's inbuilt microphone.

We now need to create an account and log in to ChatGPT (Figure 2).

Next, we need to get the API key for doing research and experimenting with the ChatGPT code, as shown in Figure 3.

You can create the API key using the right-corner option for API in your OpenAI account (Figure 4).

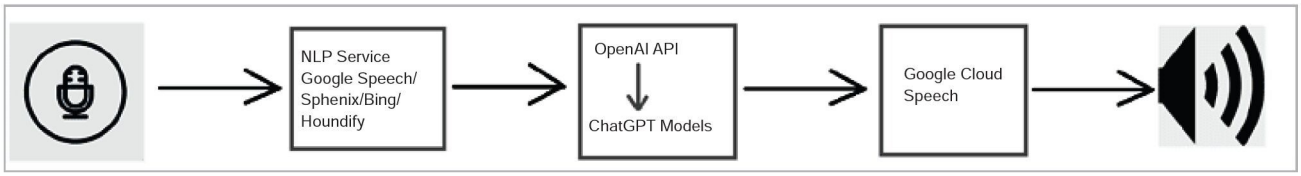



Figure 1: VoiceGPT working principle

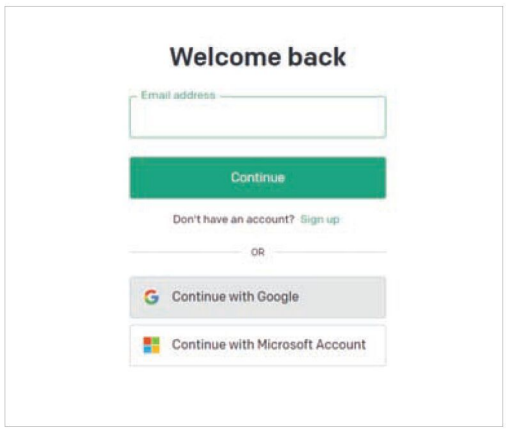



Figure 2: ChatGPT login page

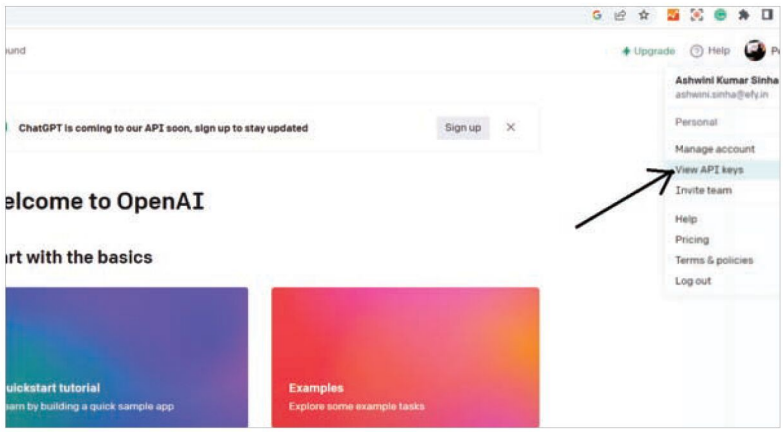



Figure 3: Getting the OpenAI API menu

After generating the OpenAI API key, copy it and save it. We need it later in our code for developing VoiceGPT.

Now we need to install the open AI on the system where we are going to run the VoiceGPT. Here you can use a computer with any Linux version installed. I used Raspberry Pi for it.

Next, open the terminal and install the open AI and other Python modules that help us in natural language processing. Here you can use Whisper from OpenAI or any other NLP module. I used Google NLP and combined it with ChatGPT.

You can install these modules using the following command. After that, you can either create your open custom talking content in OpenAI or use simple chatting in the playground. Here, you can also set the temperature frequency

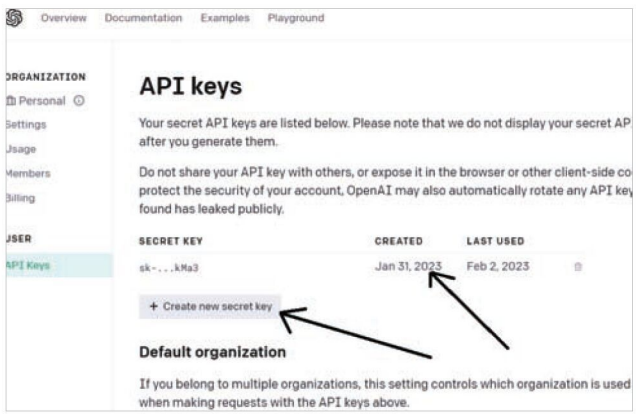



Figure 4: ChatGPT API keys

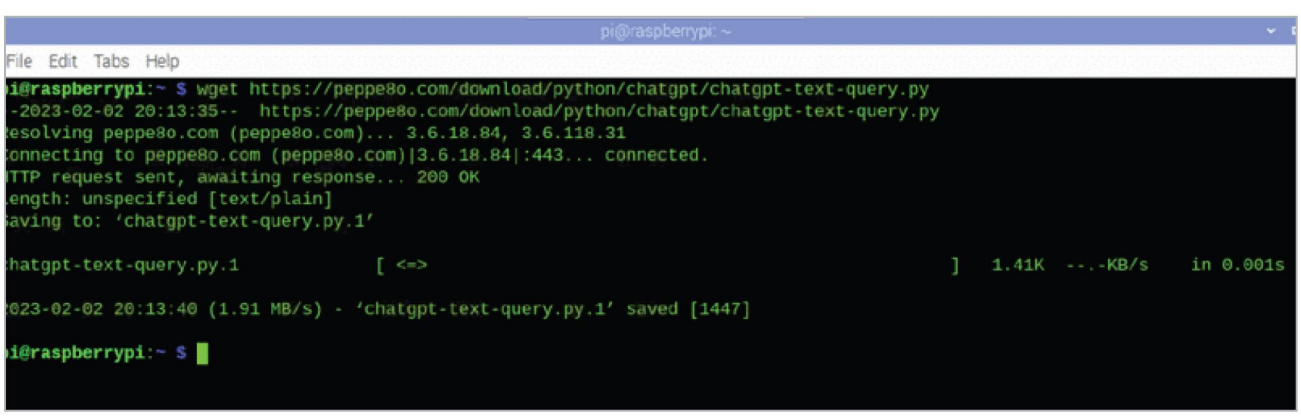



Figure 5: Cloning OpenAI ChatGPT code

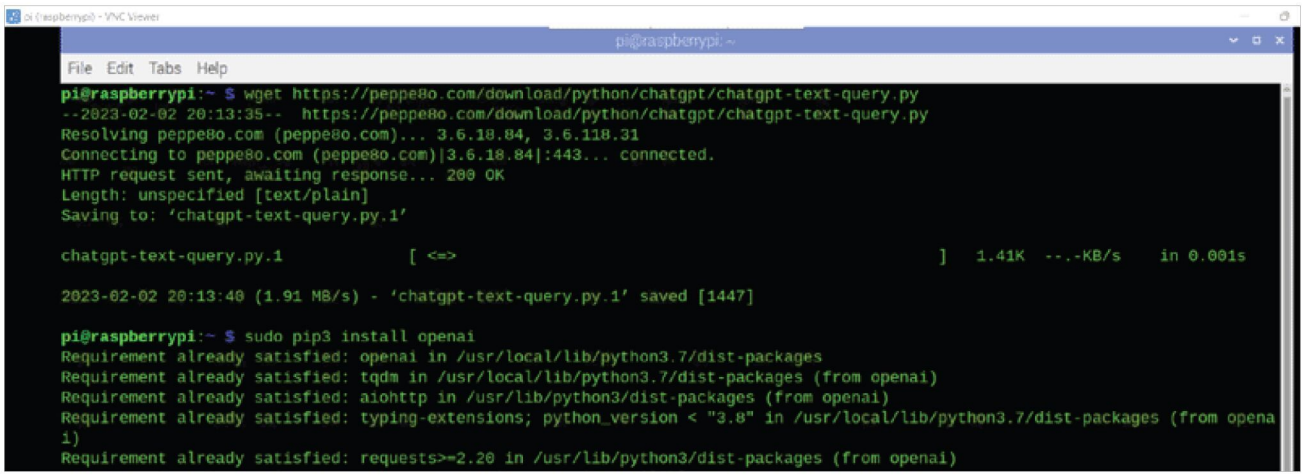


Figure 6: Raspberry Pi ChatGPT setup

and other parameters for your VoiceGPT assistant.

```
sudo pip3 install openai
sudo pip3 install SpeechRecognition
sudo pip3 install gTTS
```

Refer to Figures 5 and 6 to see how to clone the OpenAI ChatGPT and do the setup.

Next, set the temperature, frequency and chat model as shown in Figure 7.

## Programming ChatGPT to be used as VoiceGPT

First, we need to import the OpenAI Python module in code to play with OpenAI and carry out an experiment with ChatGPT. Next, we import the modules for NLP. After that, we import *pygame* to play the file that processed the reply in a human voice using the NLP model.

Next, we need to set the ChatGPT model. Here, we can choose from model names like Davinci, Ada, etc. Each model has its own expertise, and the cost of using these models varies. But no worries, because developers get a US$ 18 credit to develop and experiment with OpenAI.

Next, we need to set the API in the code. With that, we have created the function for connecting with ChatGPT to handle the query and get the response from it.

```
import speech_recognition as sr
import math
import time
import serial
from espeak import espeak
import sys
import openai
import pygame
from gtts import gTTS
pygame.mixer.init()
#model_to_use="text-davinci-003" # most capable
#model_to_use="text-curie-001"
#model_to_use="text-babbage-001"
model_to_use="text-ada-001" # lowest token cost
r = sr.Recognizer()
openai.api_key="******Your Key Here*******"
def chatGPT(query):
                response = openai. Completion.create(
                                model-model_to_use,
                                prompt=query,
                                temperature=0,
                                max_tokens 1000
                                 )
return str.strip(response['choices'][0]['text']),
response['usage']['total_tokens']
```

After that, we create the main function and then make a *while* loop. Here, we use NLP to capture the voice continuously and extract what we said using the NLP model and save it as a query. Then we transfer this query to ChatGPT and receive the response from it.

```
def main():
             print('LED is ON while button is pressed (Ctrl-C
for exit).')
             while True:
                with sr.Microphone() as source:
           r.adjust_for_ambient_noise (source)
           print("Say something!")
           audio r.listen(source)
           print("Recognizing Now....")
           command=str(r.recognize_google (audio))
           print("Google Speech Recognition thinks you said +
command)
           query=command
           (res, usage) = chatGPT (query)
```

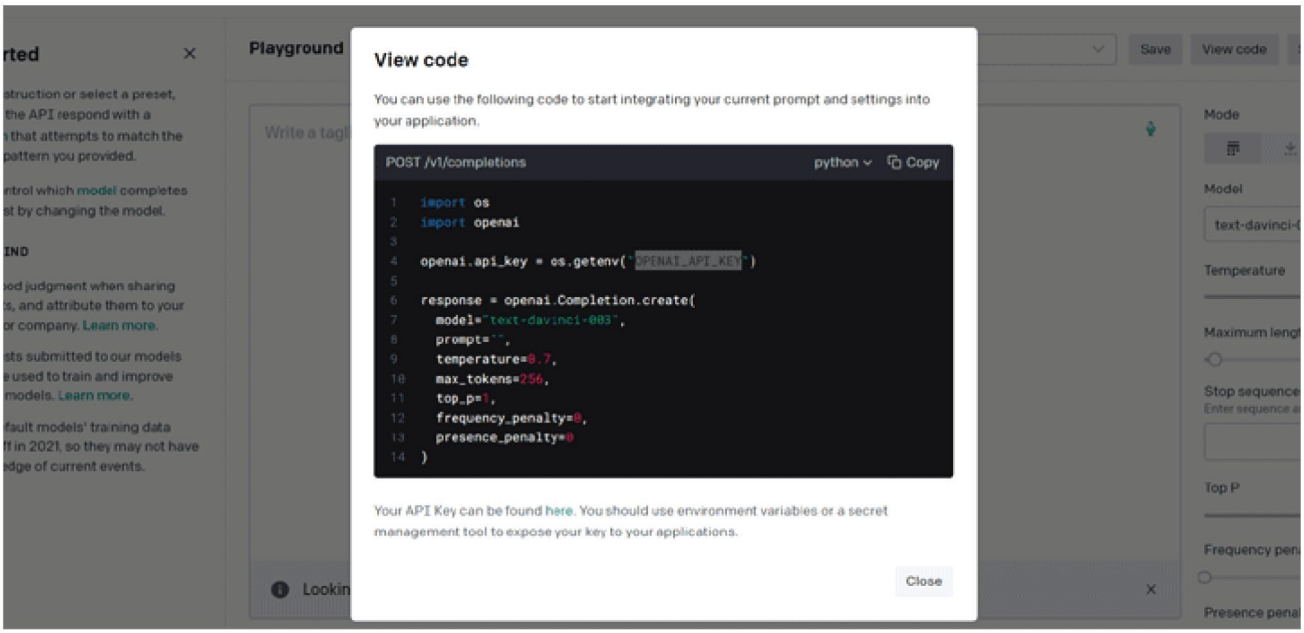



Figure 7: Setting temperature, frequency and chat model in ChatGPT

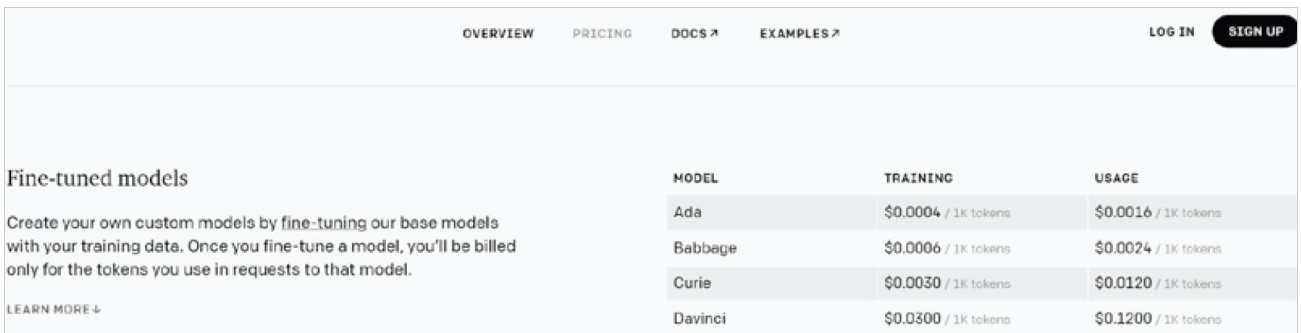



Figure 8: List of ChatGPT models

```
        print(res)
        tts gTTS(text=res, lang='en')
        tts.save("good.mp3")
        pygame.mixer.music.load("good.mp3")
        pygame.mixer.music.play()
        #espeak.synth(res)

if __name__ == '__main__':
main()
```

After this, we again use NLP to convert the reply from ChatGPT into a human voice, and then we play that voice. This whole thing runs in a loop continuously making it look like a real conversation between two humans.

This VoiceGPT gives you the option of customising and selecting models. It allows you to choose from GPT models like Ada, Davinci, or Babbage. It uses a free speech-recognition service that can be customised to offline speech-recognition services like Sphenix.

## Testing VoiceGPT

To test the VoiceGPT, run the code in Python, and it will tell you to ask a question or start a conversation. You can ask whatever you want; it recognises your voice, transfers the query to ChatGPT, and then replies to you in a human voice.

So now you can talk to ChatGPT just like you do with Google Assistant, Alexa, or Siri. Enjoy your conversation with VoiceGPT!

**Note:** This is the first version of VoiceGPT. I am still experimenting with it and you will get all the new updates very soon on *https://www.opensourceforu.com/*.

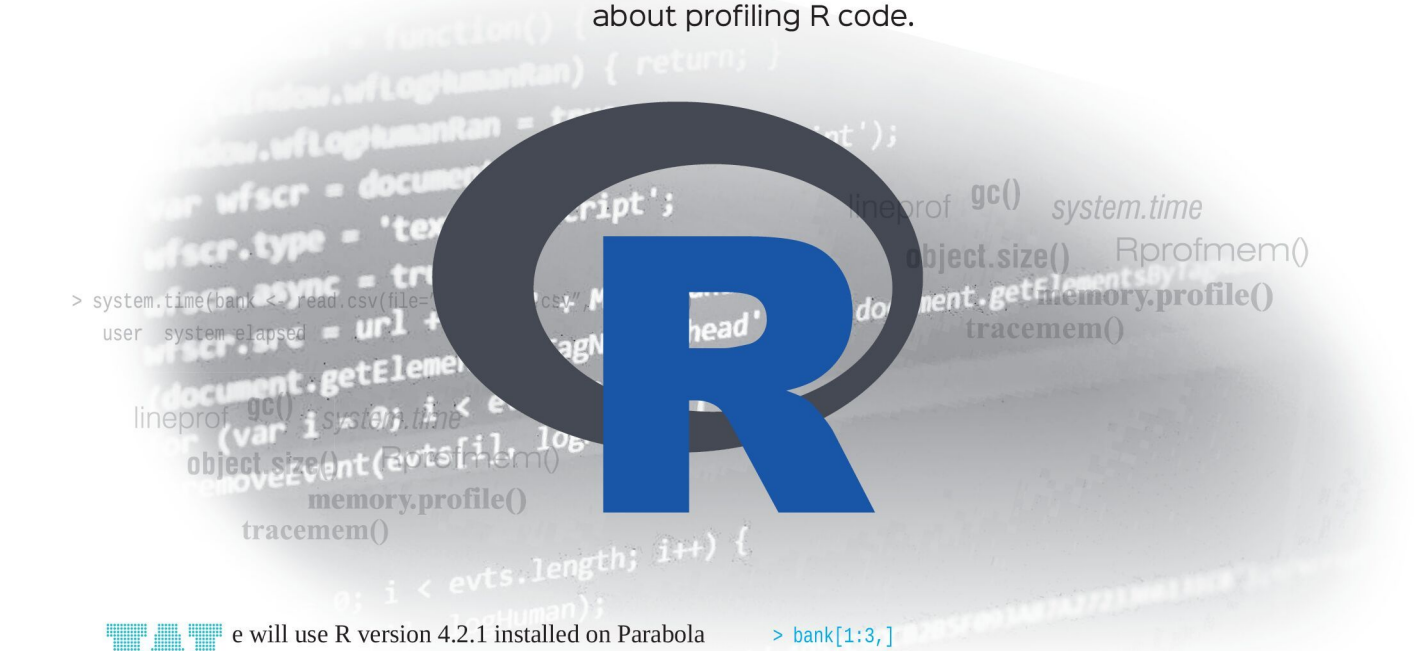
END

**By: Ashwini Kumar Sinha**

The author is a technology enthusiast at EFY.

# R Series: Profiling

In this nineteenth article in the R series, we shall learn about profiling R code.




We will use R version 4.2.1 installed on Parabola GNU/Linux-libre (x86-64) for the code snippets in this article.

```
$ R --version

R version 4.2.1 (2022-06-23) -- "Funny-Looking Kid"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under the terms of the
GNU General Public License versions 2 or 3.
For more information about these matters see
https://www.gnu.org/licenses/.
```

## system.time

Consider the 'Bank Marketing Data Set' for a Portuguese banking institution available from the UCI Machine Learning Repository available at *https://archive.ics.uci.edu/ml/datasets/Bank+Marketing*. The data can be used for public research use. There are four data sets available, and we will use the *read.csv()* function to import the data from a 'full-bank.csv' file. The *system.time()* function on an R expression provides the user, system and total time for the program, as shown below:

```
> system.time(bank <- read.csv(file="bank-full.csv", sep=";"))
   user  system elapsed
  0.129   0.000   0.135
```

```
> bank[1:3,]
  age          job marital education default balance housing
loan contact day
1  58   management married  tertiary      no    2143     yes
no unknown   5
2  44   technician  single secondary      no      29     yes
no unknown   5
3  33 entrepreneur married secondary      no       2     yes
yes unknown   5
  month duration campaign pdays previous poutcome  y
1   may      261        1    -1        0  unknown no
2   may      151        1    -1        0  unknown no
3   may       76        1    -1        0  unknown no
```

Another example is computing the cross-product of two matrices. The *rnorm()* function generates values for the normal distribution. It accepts 'n' number of observations along with values for mean and standard deviation. In the following code snippet, we are generating two matrices, 'm' and 'n', with 30 rows and 15,000 columns, and computing the cross-product between them using the *crossprod()* function. This is actually faster than calling 't(m) %*% y'.

```
> m = matrix(rnorm(30*15000, mean=0, sd=4), 30, 15000)
> n = matrix(rnorm(30*15000, mean=0, sd=4), 30, 15000)

> system.time(crossprod(m, n))
   user  system elapsed
  3.716   0.339   4.058
```

## object.size()

The *object.size()* function gives an estimate for the memory used by an object. It accepts an R object as its argument. You can also pass it a logical value for 'quote' to indicate if the result should be displayed within quotes. The 'units' argument can specify the units for printing the values. The following standards for the units are accepted – 'legacy', 'IEC', and 'SI'. A numeric data type consumes 48 bytes, and with the data there is an additional 8 bytes of overhead.

```
> object.size(numeric())
48 bytes
> object.size(2)
56 bytes
> object.size(m)
3600216 bytes
```

## Rprof and summaryRprof

The time spent by various R functions can be displayed using the *Rprof()* function. It takes the following arguments:

| Argument | Description |
|---|---|
| filename | Output filename to store results. Use NULL to stop profiling |
| append | Logical value to append results or overwrite the file |
| interval | The time interval between taking samples. Default is 0.02 |
| memory.profiling | Logical value to write memory information to output file |

You can create a 'profile.out' results file using the *Rprof()* function to profile the *crossprod()** function, as illustrated below:

```
> Rprof(filename = "profile.out", append = FALSE, interval =
0.03, memory.profiling=TRUE)
> crossprod(m, n)
   [,1]       [,2]          [,3]          [,4]          [,5]
    [1,]  1.596159e+02   -8.97348864 -7.103962e+01
3.863508e+01 -1.168697e+02
    [2,] -3.959816e+01   13.93104995 -7.210794e+01
3.495264e+01  4.697674e+01
    [3,]  1.329374e+02  -80.40730081  5.745854e+01
-7.841128e+01 -8.069322e+01
    [4,]  7.727353e+00   78.42727385  1.142679e+01
-9.473472e+01 -4.733995e+01
    [5,]  1.341396e+02  -50.23056812  7.478321e+01
-5.869277e+01 -9.588495e+00
    [6,]  1.741993e+01   45.03771290  5.569390e+01
-2.684412e+01  5.295564e+01
...
> Rprof(NULL)
```

After the profiling is stopped by using the 'Rprof(NULL)' command, the results can be viewed using the *summaryRprof()* function:

```
> summaryRprof(filename = "profile.out")
$by.self
            self.time self.pct total.time total.pct
"crossprod"      3.96      100       3.96       100

$by.total
            total.time total.pct self.time self.pct
"crossprod"       3.96       100      3.96      100

$sample.interval
[1] 0.03

$sampling.time
[1] 3.96
```

Since we are only running the 'crossprod()' function, the output shows that it takes all the time in the computation. The 'summaryRprof()' function accepts the following arguments:

| Argument | Description |
|---|---|
| filename | The file generated by *Rprof()* |
| memory | Memory information |
| lines | Line information |
| index | Stack trace memory information |
| diff | Logical value to indicate the difference in memory |
| exclude | Any functions to filter out in the output |

The *memory="stat"* option to the *summaryRprof()* function displays the maximum memory usage, as shown below:

```
> summaryRprof(filename = "profile.out", memory="stat")
index: "crossprod"
     vsize.small  max.vsize.small      vsize.large  max.
vsize.large
           11749          1597864         13309000
1799964112
```

```
          nodes        max.nodes     duplications tot.
duplications
         113647         15456000                0
0
        samples
            136
```

The time series memory information can be printed using the *memory="tseries"* option, as follows:

```
> summaryRprof(filename = "profile.out", memory="tseries")
     vsize.small vsize.large    nodes duplications
stack:2
0.03     1597864    10059904 15456000        0 "crossprod"
0.06           0  1799964112        0       0 "crossprod"
0.09           0           0        0     0 "crossprod"
0.12           0           0        0     0 "crossprod"
0.15           0           0        0     0 "crossprod"
0.18           0           0        0     0 "crossprod"
0.21           0           0        0     0 "crossprod"
...
```

The contents of the *profile.out* file are provided for reference:

```
$ cat profile.out

memory profiling: sample.interval=30000
:199733:1257488:15456000:0:"crossprod"
:199459:226253002:15412208:0:"crossprod"
:199459:226253002:15412208:0:"crossprod"
:199459:226253002:15412208:0:"crossprod"
:199459:226253002:15412208:0:"crossprod"
...
```

## memory.profile()

The *memory.profile()* function provides memory usage information based on the object type. For example:

```
> memory.profile()
     NULL    symbol   pairlist   closure environment   promise
        1      6943     139922      3134         970      5923
  language   special   builtin      char     logical   integer
     35223        47       697      8089        5501     24059
    double   complex   character     ...         any      list
      1085         1     39918         0           0     11122
expression  bytecode externalptr   weakref     raw        S4
         1      8843       976      1166         399       917
```

## Rprofmem()

A more detailed memory profiling can be accomplished using the *Rprofmem()* function. It accepts the following arguments:

| Argument | Description |
|---|---|
| filename | The file to write output results |
| append | Logical value to indicate to append or overwrite the file |
| threshold | Allocations larger than R's large vector heap will be reported |

You can initialise 'Rprofmem()' and then execute the 'crossprod()' function as illustrated in the example below. The 'Rprofmem(NULL)' command is used to stop the profiling. The contents of the output file show the memory used by the 'crossprod()' function.

```
> Rprofmem("Rprofmem.out", threshold=1000)
> crossprod(m, n)
                 [,1]          [,2]          [,3]
[,4]          [,5]
    [1,]  -33.86416785 -6.161515e+01  1.376119e+02
-1.494083e+01  5.784715e+01
    [2,]  -59.34911436  6.868712e+01 -1.899812e+01
-6.986599e+01 -1.165873e+02
    [3,] -151.68100670 -7.330638e+01 -1.320967e+02
9.607252e+01  1.459454e+02
    [4,]  132.50907447  4.157651e+01  7.621391e+01
-4.062714e+00 -9.967460e+01
    [5,]  -53.53606072  5.132813e+01 -3.571266e+01
-2.835573e+00  6.216842e+01
    [6,]   41.33625436 -1.246898e+02  1.150376e+02
-7.413852e+01 -9.574722e-03
    ...

> Rprofmem(NULL)

$  cat Rprofmem.out
1800000048 :"crossprod"
60056 :
...
```

## lineprof

The *lineprof()* built-in is used to provide profiling information for every line in an R program. You can install the same using the following steps:

```
> install.packages("devtools")

> library(devtools)
```

```
Loading required package: usethis

> devtools::install_github("hadley/lineprof")
Downloading GitHub repo hadley/lineprof@HEAD
...
 ** testing if installed package can be loaded from temporary
location
 ** checking absolute paths in shared objects and dynamic
libraries
 ** testing if installed package can be loaded from final
location
 ** testing if installed package keeps a record of temporary
installation path
 * DONE (lineprof)
```

Consider a function 'f' that creates two matrices, doubles its contents, and computes their cross-product. The usage of the 'lineprof' function is shown below:

```
> f <- function() {
+   m = matrix(rnorm(30*15000, mean=0, sd=4), 30, 15000)
+   n = matrix(rnorm(30*15000, mean=0, sd=4), 30, 15000)
+   a <- m * 2
+   b <- n * 2
+   crossprod(a, b)
+ }

> l <- lineprof(f())

> l
   time   alloc release dups                 ref
src
1 0.005   2.151   0.000    0 c("matrix", "rnorm") matrix/
rnorm
2 0.410 214.577   0.878    0         "crossprod" crossprod
```

We again observe that the 'crossprod()' function takes the maximum memory allocation and computation time.

## gc()

The *gc()* function triggers the garbage collection to free memory. It also displays information on available memory, as shown below:

```
> gc()
         used (Mb) gc trigger   (Mb)  max used   (Mb)
Ncells  294146 15.8     661162   35.4    464793   24.9
Vcells 2457665 18.8  220002509 1678.5 227627845 1736.7
```

## tracemem()

The *tracemem()* function is used to debug memory usage issues in R. It can print a message whenever a marked object for tracing is copied. In the following example, we create an 'a' matrix and track its memory usage with the 'tracemem' function. The matrix is copied to a 'b' matrix, and its contents are then modified. This triggers a printing of the memory address in the output. You can finally stop tracing an R object with the *untracemem()* function, as shown below:

```
> a <- matrix(1:8, nrow=2)
> a
     [,1] [,2] [,3] [,4]
[1,]    1    3    5    7
[2,]    2    4    6    8

> tracemem(a)
[1] "<0x563f0483c808>"

> b <- a

> b[1] <- 10
tracemem[0x563f0483c808 -> 0x563f04838078]:
tracemem[0x563f04838078 -> 0x563f04c850e8]:

> b

     [,1] [,2] [,3] [,4]
[1,]   10    3    5    7
[2,]    2    4    6    8

> a

     [,1] [,2] [,3] [,4]
[1,]    1    3    5    7
[2,]    2    4    6    8

> untracemem(a)
```

You are encouraged to read the help pages of the above R functions to learn more about their arguments, options and usage. END

### References


- [Moro et al., 2014] S. Moro, P. Cortez and P. Rita, A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014
- Hadley Wickham. Memory. Advanced R, *http://adv-r.had.co.nz/memory.html*



**By: Shakthi Kannan**

The author is a free software enthusiast.

# Building a Quiz Service with Spring Boot

This DIY project demonstrates the use of Spring Boot to develop a simple quiz service. The service exposes a REST API and a set of random questions, and generates a quiz with a set of ten questions.

Spring Boot is a Java framework for developing enterprise applications. With several sub-frameworks like Spring Data, Spring Rest, etc, Spring Boot helps in the quick development of full-stack applications. This DIY project demonstrates developing the server side of a Quiz application using Spring Boot.

The objective of the project is to develop the server side of a simple quiz application and expose it through REST API.

The API requirements are:

- An API to upload quiz questions. Each question consists of four options with one of them being the correct option. Each question belongs to a specific topic under a specific subject.
- An API to generate a quiz with a set of ten random questions on a specific subject and topic.
- An API to submit answers and get the score.

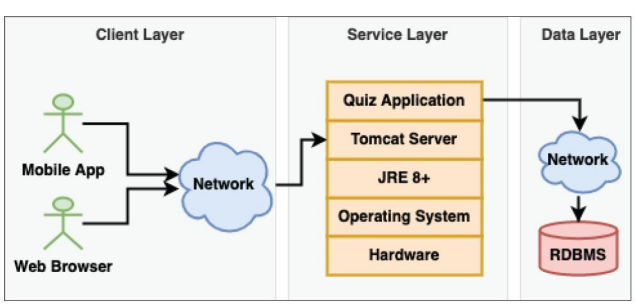



Figure 1: Architecture

The architectural requirements are as follows.

*Database:* Any RDBMS like MySQL

*Environment:* Spring Boot framework on Java 8+ platform with an embedded Tomcat server

The Quiz application represents a typical three-layer Java web application. The scope of this project is limited only to the service and data layers.

## REST API

The Quiz service exposes the following three REST endpoints.

1. *POST /question* adds a new question to the system. Successful addition of the question returns the HTTP 201 status. The request payload will be a JSON with the following model:

```
{
    description: string,
    optionOne: string,
    optionTwo: string,
    optionThree: string,
    optionFour: string,
    answer: int,
    subject: string,
    topic: string
}
```

The response payload will be a JSON with the following model:

```
{
    qid: int,
    description: string,
    optionOne: string,
    optionTwo: string,
    optionThree: string,
    optionFour: string
}
```

2. *GET /questions?subject=SUBJECT&topic=TOPIC* gets a set of ten random questions from the specified SUBJECT

and TOPIC. The response payload will be a JSON array of the following:

```
{
    qid: int,
    description: string,
    optionOne: string,
    optionTwo: string,
    optionThree: string,
    optionFour: string
}
```

3. *POST /answers* submits the answers and gets the score. The request payload will be a JSON array of the following:

```
{
    qid: int,
    option: int
}
```

And the response payload will be a JSON with the following structure:

```
{
    total: int,
    rights: int
}
```

## Domain model and design

The domain model consists of the Question entity. The field *qid* represents the identity and it is system generated. The field's subject, topic and answer will not have any getters and setters as they are hidden from the users of the system.

The repository will be an interface extending the JpaRespository interface, as the Question objects are stored in an RDBMS system and accessed using Java Persistence API. The repository is extended with three custom methods in line with JPQL.

The controller is a REST controller with appropriate HTTP mappings.

## Implementation

***Step 1: Create a Spring project with Maven support***

Choose Spring version 2.7.3, which is the latest version that supports Java 8+. Add *spring-boot-starter-web* for REST support, and *spring-boot-starter-data-jpa* for JPA-based database connectivity, apart from other dependencies for JSON bindings, database drivers, etc.

Given below is the extract of the critical dependencies in the *pom.xml*. Observe that we are using an H2 database in this project, which can be replaced with any other RDBMS.

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-core</artifactId>
</dependency>
<dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-databind</artifactId>
</dependency>
<dependency>
```

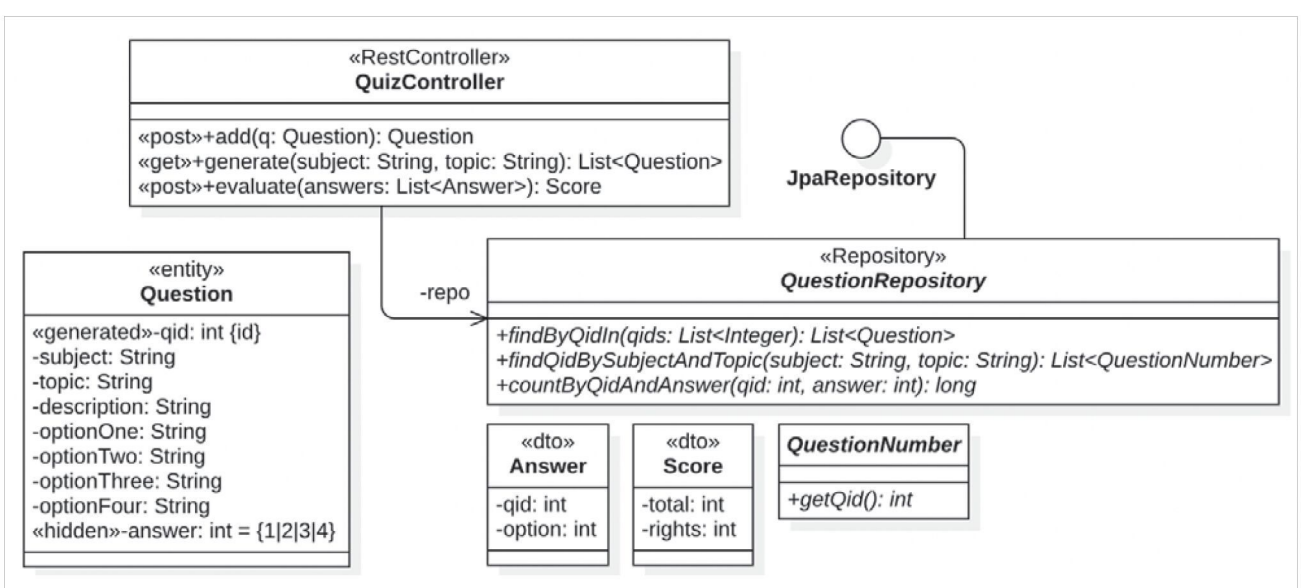



Figure 2: Class model

```
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>
<dependency>
    <groupId>com.h2database</groupId>
    <artifactId>h2</artifactId>
    <scope>runtime</scope>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-devtools</artifactId>
    <scope>runtime</scope>
    <optional>true</optional>
</dependency>
```

***Step 2: Develop the domain classes***

This project has only one domain class, namely, Question, which is mapped as a JPA entity.

```
@Entity
public class Question {
    @Id
    @GeneratedValue
    private int qid;
    private String description;
    private String optionOne;
    private String optionTwo;
    private String optionThree;
    private String optionFour;
    private int answer;
    private String subject;
    private String topic;

    // add the constructors

    public int getQid() {
      return qid;
    }

    public String getDescription() {
      return description;
    }

    public String getOptionOne() {
      return optionOne;
    }

    public String getOptionTwo() {
      return optionTwo;
    }

    public String getOptionThree() {
      return optionThree;
    }

    public String getOptionFour() {
      return optionFour;
    }
}
```

It does not offer any setters, as the questions are not expected to be updated once added. Also, since the Question objects will be returned to the REST clients, hide the subject, topic and answer fields by not exposing the corresponding getters.

***Step 3: Develop the data layer***

The repository follows the JPQL and Spring Data conventions.

```
@Repository
public interface QuestionRepository extends
JpaRepository<Question, Integer> {
    List<Question> findByQidIn(List<Integer> qids);

    List<QuestionNumber> findQidBySubjectAndTopic(String
subject, String topic);

    long countByQidAndAnswer(int qid, int option);
}
```

The *findByQidIn()* method fetches the questions with the supplied list of Question IDs.

The *findQidBySubjectAndTopic()* method fetches the list of IDs of all the questions belonging to the specified subject and topic. The QuestionNumber interface is used by this method, as the operation involves projection.

```
public interface QuestionNumber {
    public int getQid();
}
```

The *countByQidAndAnswer0* method returns 1 if the supplied option matches the expected answer; otherwise, it returns 0.

***Step 4: Develop the REST layer***

This layer consists of controllers and DTOs. The QuizController with the @RestController annotation maps the REST endpoints to the methods and processes the HTTP requests with the help of injected QuestionRepository.

```
@RestController
public class QuizController {
     @Autowired
     private QuestionRepository repo;

     // mapping methods

}
```

The *add()* method maps *POST /question* API, accepts Question as RequestBody of the *HTTP Post* request, and saves it in the repository and thereby in the database.

```
     @PostMapping("/question")
     public ResponseEntity<Question> add(@RequestBody
Question question) {
       question = repo.save(question);
       return ResponseEntity.status(HttpStatus.CREATED).
body(question);
     }
```

The *generate()* method maps *GET /questions* API, extracts subject and topic parameters from the request URI, and fetches the matching question IDs from the repository. It randomly picks ten question IDs and returns the corresponding questions. Randomisation can also be done in the database itself, but not all databases support such an operation. To make the implementation work for any RDBMS, we chose to randomise the question IDs in the service layer.

```
     @GetMapping("/questions")
     public ResponseEntity<List<Question>> generate(@
RequestParam("subject") String subject,
          @RequestParam("topic") String topic) {
       List<Integer> ids = repo.findQidBySubjectAndTopic(subje
ct, topic).stream().map(qn -> qn.getQid())
              .collect(Collectors.toList());
       Collections.shuffle(ids);
       List<Question> questions = repo.findByQidIn(ids.
subList(0, 10));
       return ResponseEntity.status(HttpStatus.OK).
body(questions);
     }
```

The *evaluate()* method maps *POST /answers* API and gets the list of answers from the request body. For each of the answers, the method checks with the repository if it is right or wrong, and returns the final computed score.

```
     @PostMapping("/answers")
     public ResponseEntity<Score> evaluate(@RequestBody
List<Answer> answers) {
       int rights = 0;
       for (Answer answer : answers)
          rights += (int) repo.countByQidAndAnswer(answer.
getQid(), answer.getOption());
       Score score = new Score(answers.size(), rights);
       return ResponseEntity.status(HttpStatus.CREATED).
body(score);
     }

}
```

The Answer and Score objects are simple POJOs.

```
public class Answer {
     private int qid;
     private int option;

     ...
}

public class Score {
     private int total;
     private int right;

     ...
}
```

***Step 5: Configuration***

The *application.properties* provides the configuration for the data source. Since we are using H2 in this project, the following configuration works for it:

```
spring.datasource.url=jdbc:h2:mem:quiz-db
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.username=sa
spring.jpa.properties.hibernate.dialect=org.hibernate.
dialect.H2Dialect
spring.jpa.hibernate.ddl-auto= update
spring.h2.console.enabled=true
```

The Tomcat server runs on the default 8080 port. We added configuration to add */quiz* as the URL prefix.

```
server.port=8080
server.servlet.contextPath=/quiz
```

The complete code of the application is available at *https://bitbucket.org/glarimy/glarimy-university/src/master/glarimy-boot/*.

***Step 6: Build, run, test, and deploy***

Build the application and access the database at *http://*

*localhost:8080/quiz/h2-console/* with *jdbc:h2:mem:quiz-db* as the URL string. You can find that the tables are created automatically.

Use tools like Postman or cURL to add questions. For instance, the following *curl* command adds a new question.

```
curl -X 'POST' 'http://localhost:8080/quiz/question' -H
'Content-Type: application/json' -d '{"description":
"What is the capital of India?", "optionOne":
"Bengaluru", "optionTwo": "New Delhi", "optionThree":
"Mumbai", "optionFour": "Kolkota", "subject":"Politics",
"topic":"India", "answer": 2}'
```

Add as many questions as possible. The following *curl* command generates a new quiz with ten questions.

```
curl 'http://localhost:8080/quiz/questions?subject=Politics&
topic=India
```

Repeat the above call several times. You will find different questions being generated, provided that there are a good number of questions in the database.

Submit the answers using the *curl* command, which looks like the following:

```
curl -X 'POST' 'http://localhost:8080/quiz/answers' -H
'Content-Type: application/json' -d '[{"qid": 1, "option":
1},{"qid": 10, "option": 2},{"qid": 4, "option": 2},{"qid":
6, "option": 2},{"qid": 14, "option": 1},{"qid": 25,
"option": 4},{"qid": 7, "option": 1},{"qid": 18, "option":
3},{"qid": 3, "option": 4},{"qid": 19, "option": 1}]'
```

The response will report the score.

## Extension

This simple application can be extended with several enhancements. You can:

1. Generate a quiz with a specified number of questions, instead of fixing the size just to ten.
2. Add authentication and authorisation in such a way that only the admin is able to call the *POST /question* API and only students are able to call the other two APIs.
3. Store the quiz and scores so that they can be retrieved later by the students and admin.
4. Add APIs to list the available subjects and topics.
5. Add a feature to generate and store the quiz so that it can be accessed by students at a specified time and for a specified duration. END

**By: Krishna Mohan Koyya**

The author is the founder and principal consultant at Glarimy Technology Services, Bengaluru. He has mentored and upskilled more than 250 technical teams in architecting, designing, and developing enterprise applications on-premises as well as on cloud.

Statement about ownership and other particulars about
**OPEN SOURCE FOR YOU**
FORM IV (See Rule 8)

| | | |
|---|---|---|
| 1. Place of publication | : | New Delhi |
| 2. Periodicity of its publication | : | Monthly |
| 3. Printer's Name | : | Ramesh Chopra |
| Nationality | : | Indian |
| Address | : | OPEN SOURCE FOR YOU<br>D-87/1, Okhla Industrial Area,<br>Phase-1, New Delhi 110020 |
| 4. Publisher's Name<br>Nationality<br>and Address | : | Same as (3) above |
| 5. Names and addresses of individuals who own the newspaper & partners or shareholders holding more than 1% of the total capital | : | **EFY Enterprises Pvt Ltd.**<br>D-87/1, Okhla Industrial Area,<br>Phase-1, New Delhi 110020 |

I, Ramesh Chopra, hereby declare that the particulars given above are true to the best of my knowledge and belief.

Date: 28-2-2023

# Integrating Network Function Virtualization with the DevOps Pipeline: Kubernetes

The third part of this series of articles on integration of network function virtualization with the DevOps pipeline discusses the architecture of Kubernetes and the various ways to build a cluster for your infrastructure.


Kubernetes is open source container orchestration software. With Kubernetes, users can run multiple containers over many different machines and automate the life cycle of containers over such distributed systems. Kubernetes can automate application deployment, scale the size of the application, and manage the containers. It can scale the application using automation without increasing the size of the operations team. In short, it is container-centric management software.

Kubernetes is often known as the kernel of distributed systems because it abstracts the mounting hardware of the nodes away from the application that is running. It provides a common ground for the workload deployed and the applications running to consume the shared pool of resources. This handling of resources by Kubernetes eases the life cycle of deployment of a large scale containerized application.

Some of the features of Kubernetes include:

- Automated rollouts and rollbacks
- Service discovery and load balancing
- Storage orchestration
- Secret and configuration management
- Horizontal scaling
- Self-healing

## Architecture

When we talk about Kubernetes architecture, there are two components to look out for — master and worker nodes. These nodes follow the master/slave architecture. The master node

is also known as the control plane. Applications are deployed over worker nodes, specifically in pods. Pods are one or more containers that share volumes and a network name space, and are part of a single context.

The control plane is the commanding node of the Kubernetes cluster. It communicates with all the worker nodes of the cluster. It manages the various services of the worker nodes and makes global decisions about the cluster. The components of the control plane are listed below.

**kube-apiserver**: This exposes the Kubernetes API. The API server is the front end of the Kubernetes control plane.

**etcd**: This is a distributed, consistent key-value store used for configuration management, service discovery, and coordinating distributed work.

**kube-scheduler**: This watches for newly created pods with no assigned node, and binds them to a node to run on.

**kube-controller-manager**: This component manages all core components and makes the necessary changes in an attempt to move the current state towards the desired state.

**cloud-controller-manager**: This lets you link your cluster to your cloud provider's API.

The worker node is responsible for managing the running pods and enabling Kubernetes container runtime. Each node in the cluster runs a container runtime and the following components.

**kubelet**: This is responsible for running your application and reporting the status of the node to the API server in the master node.

**kube-proxy**: This runs on each node and is responsible for watching the API server for any changes while maintaining the entire network configuration up to date.

The Kubernetes cluster shown in Figure 1 outlines all the components

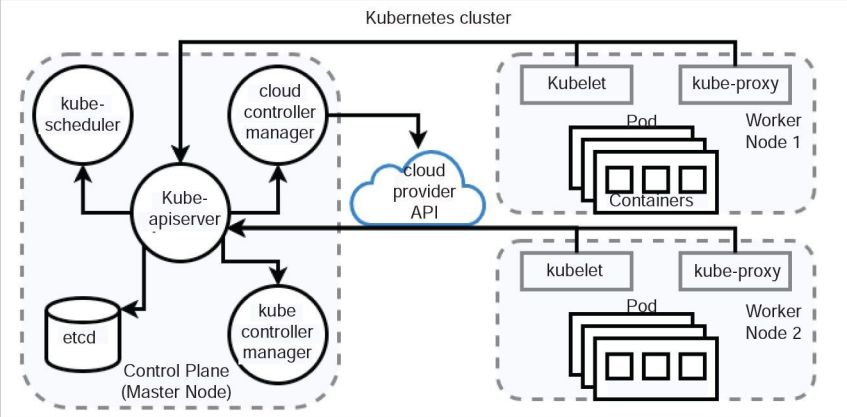



Figure 1: Kubernetes cluster architecture with manager node and worker nodes

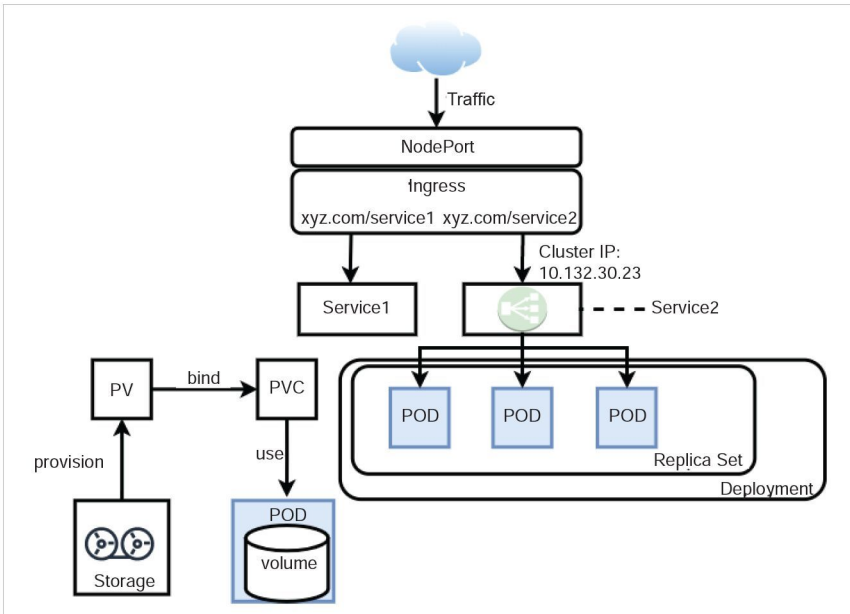



Figure 2: Synchronization and interaction of the various components of Kubernetes worker node

involved. The master node connects to the worker nodes to produce an environment to facilitate all the application pods. Kubelet in worker nodes reports the status via the exposed APIs of the master node while serving the pods within.

Some of the other important components representing the workload, network, and storage of a Kubernetes cluster are given below.

**Pod**: This is the smallest unit of work or management resource within Kubernetes. It is composed of one or more containers that share their storage, network, and context.

**ReplicaSets**: This is a method of managing pod replicas and their life cycle. It keeps the specified number of pods constant and is responsible for scheduling, scaling, and deletion of pods.

**Deployment**: This is a declarative method to manage various pods and their replica sets. With deployment, users can describe the life cycle of applications and images.

**Cluster**: The resources of this set of nodes are used to run a set of applications.

**Jobs**: These are best used to run a finite task to completion as opposed

to managing an ongoing desired application state.

**Service**: This exposes an application running on a set of pods as a network service.

**ClusterIP**: This component exposes the service on a cluster-internal IP. Choosing this value makes the service reachable only from within the cluster (default).

**NodePort**: This exposes the service on each node's IP at a static port (the NodePort). A ClusterIP service, to which the NodePort service will route, is automatically created.

**Ingress**: This is the primary method of exposing a cluster service to the outside world. These are load balancers or routers that usually offer secure sockets layer (SSL) termination, name-based virtual hosting, etc.

**Volume**: This is storage that is tied to the pod life cycle, consumable by one or more containers within the pod.

**Persistent volume (PV)**: This represents storage resources in the cluster. It is the 'physical' volume on the host machine that stores your persistent data.

**PersistentVolumeClaim (PVC)**: This fulfills the request for storage by the user, where it claims the resources from the PersistentVolume.

Figure 2 depicts the flow of the request to the deployed application over Kubernetes. The various components of Kubernetes come together to facilitate an environment for the services. The NodePort exposes the static port at which the traffic is intercepted, and based on the request the Ingress segregates it. Ingress filters the request and forwards it to the appropriate service, which is sitting on the top of deployment in the Kubernetes cluster. A deployment can have multiple replicas of a pod, which is responsible for the business logic and data retrieval/modification. Data-based services are handled by the volume associated with the pod. The cluster administrator provisions the infrastructure storage.

## Kubernetes container runtime

Kubernetes is attached to a container runtime, which is responsible for the life cycle of the containers on Kubernetes nodes while also managing container images and their pods. Container runtime is also responsible for container interactions such as attach, exec, ports, logs, etc. With the release of Kubernetes version 1.5, the Kubernetes community introduced the Container Runtime Interface (CRI). This is a new plugin API for container runtime in Kubernetes which connects container runtime with kubelet.

With its implementation as a separate entity, it allows users to switch out container runtime implementations instead of having them built into the kubelet. As shown in Figure 3, kubelet communicates with the container runtime; it bridges that communication via the use of gRPC (Google remote procedure call), where kubelet acts as a client and the CRI shim acts as the server. A shim is a library that intercepts various API calls and then either redirects the operations elsewhere or handles them itself. In this routine, the shim can change the passed arguments and facilitates communication. Some of the common container runtimes with Kubernetes are containerd, CRI-O, and Docker.

## Setting up a Kubernetes cluster using minikube

minikube is a locally deployed single-node Kubernetes cluster environment that can be used as a learning or development environment. It can be

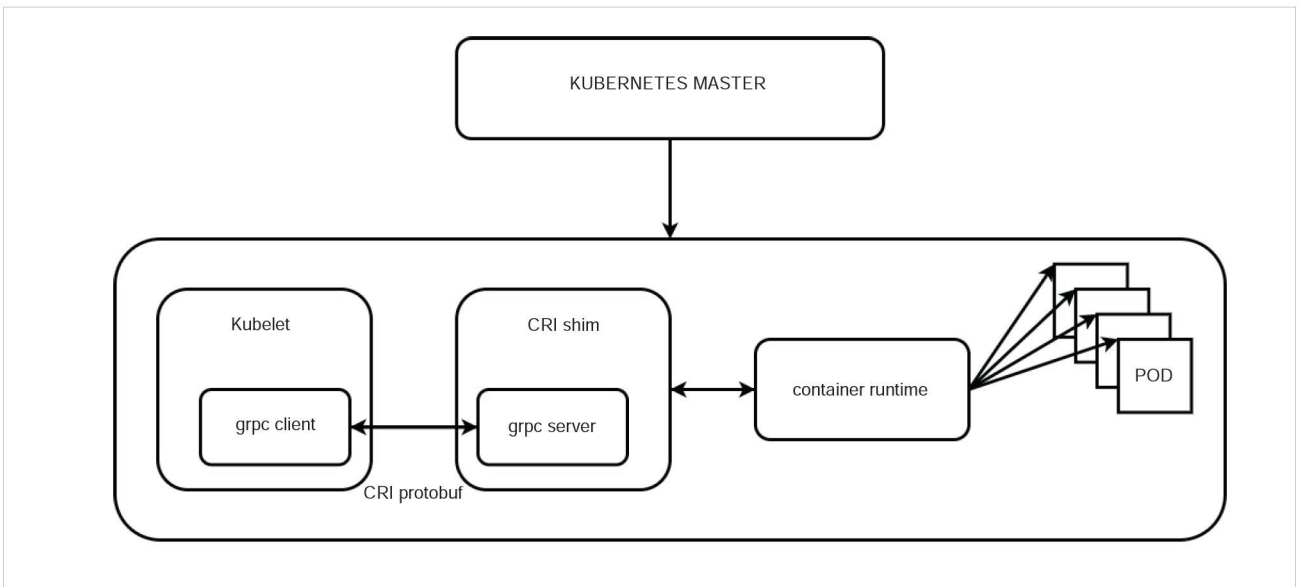



Figure 3: Various components of container runtime in Kubernetes worker node

```
shubham@k8s-minikube:~$ minikube kubectl -- get nodes -o wide
NAME       STATUS   ROLES                  AGE   VERSION   INTERNAL-IP    EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION      CONTAINER-RUNTIME
minikube   Ready    control-plane,master   5m42s   v1.23.3   192.168.49.2   <none>        Ubuntu 20.04.2 LTS   5.4.0-109-generic   docker://20.10.12
shubham@k8s-minikube:~$
```

Figure 4: Kubernetes node in minikube cluster with default container runtime

```
shubham@k8s-minikube:~$ minikube kubectl -- get pods --all-namespaces -o wide
NAMESPACE     NAME                               READY   STATUS    RESTARTS        AGE     IP              NODE       NOMINATED NODE   READINESS GATES
kube-system   coredns-64897985d-4mkd9            1/1     Running   0               4m58s   172.17.0.2      minikube   <none>           <none>
kube-system   etcd-minikube                      1/1     Running   0               5m19s   192.168.49.2    minikube   <none>           <none>
kube-system   kube-apiserver-minikube            1/1     Running   0               5m19s   192.168.49.2    minikube   <none>           <none>
kube-system   kube-controller-manager-minikube   1/1     Running   0               5m13s   192.168.49.2    minikube   <none>           <none>
kube-system   kube-proxy-zjbtl                   1/1     Running   0               4m58s   192.168.49.2    minikube   <none>           <none>
kube-system   kube-scheduler-minikube            1/1     Running   0               5m19s   192.168.49.2    minikube   <none>           <none>
kube-system   storage-provisioner                1/1     Running   1 (4m22s ago)   5m6s    192.168.49.2    minikube   <none>           <none>
shubham@k8s-minikube:~$
```

Figure 5: All the pods in Kubernetes minikube

started and stopped like any other Linux service. minikube can run components of Kubernetes such as pod, Ingress, service, and others. For minikube to be deployed the machine must have either container or virtual machine managers such as Docker, Hyperkit, HyperV, Kernel-based Virtual Machine (KVM), Parallels, Podman, VirtualBox, or VMware Fusion/Workstation installed on the host machine.

The local installation was done on an Ubuntu 20.04.2 LTS virtual machine, with two CPUs, 8GB memory, and 50GB storage.

Next, we updated and upgraded the target platform:

```
$ sudo apt update
$ sudo apt upgrade
```

The following commands get downloaded and install the latest minikube stable release on the target platform:

```
$ curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
$ sudo install minikube-linux-amd64 /usr/local/bin/minikube
```

And that's it; Kubernetes is successfully installed over the node. To start the Kubernetes cluster, run the following command:

```
$ minikube start
```

Figure 4 shows all the nodes attached to the Kubernetes cluster and proves the point that minikube is a single-node Kubernetes cluster that acts as a control plane (master) node under ROLES. The STATUS tells whether the node is active or inactive. The AGE tells the user how long the node has been active. VERSION depicts the version of the minikube installation.

The minikube cluster is connected to a network, and the network settings give an IP address to the node, which is read as INTERNAL-IP. No EXTERNAL-IP is attached to the node; that's why we see none here. As mentioned earlier, the installation is done over an Ubuntu-based system, and the OS-IMAGE showcases the same. In KERNEL-VERSION we see the version of the kernel of the OS platform. Finally, minikube uses the container runtime to achieve the life cycle of the containers within the cluster, and CONTAINER-RUNTIME shows which version of container runtime it uses internally as well as its version.

Figure 5 shows all the pods attached to all the name spaces in the minikube cluster. Under NAMESPACES, all the name spaces attached to the minikube cluster are visible. NAME showcases the names of all the pods that are running. READY demonstrates the number of pods running as against the number of pods required. All the pods are running, which is evident from STATUS. If there is any restart in any particular pod, that can be seen under RESTARTS. minikube creates an internal subnet, and some pods are connected to it, while other pods are connected to the internal IP address of the cluster. NODE shows which place each pod is running at. Since we went with the default settings, there was no priority set between nodes -- that's why NOMINATED NODE shows none. READINESS GATES helps implement custom checks for any running pod. Since we went with default settings, it shows none.

## Setting up a Kubernetes cluster using KVM

A Kubernetes cluster consists of more than a single node (minikube). The best practice is to have multiple worker nodes to enable high availability. This Kubernetes cluster is managed by the user via the kubeadm tool. Container runtime is deployed over each node of the cluster so that it manages the life cycle of the pods. We'll learn how to facilitate a Kubernetes cluster over multiple KVM nodes (master and workers) with common container runtimes such as:

- CRI-O
- Docker Engine
- Containerd

The setup of the Kubernetes cluster was done on two Ubuntu 20.04.4 LTS virtual machines. Here the master and worker node configurations were two CPUs, 4GB of memory, and 50GB of storage.

```
shubham@k8s-crio-master:~$ kubectl get nodes -o wide
NAME              STATUS   ROLES                  AGE   VERSION   INTERNAL-IP      EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION      CONTAINER-RUNTIME
k8s-crio-master   Ready    control-plane,master   6m3s  v1.23.6   192.168.122.21   <none>        Ubuntu 20.04.4 LTS   5.4.0-109-generic   cri-o://1.23.2
k8s-crio-worker   Ready    <none>                 103s  v1.23.6   192.168.122.22   <none>        Ubuntu 20.04.4 LTS   5.4.0-109-generic   cri-o://1.23.2
shubham@k8s-crio-master:~$
```

Figure 6: Kubernetes nodes with CRI-O runtime

```
shubham@k8s-crio-master:~$ kubectl get pods --all-namespaces -o wide
NAMESPACE     NAME                                          READY   STATUS    RESTARTS   AGE     IP               NODE              NOMINATED NODE   READINESS GATES
kube-system   coredns-64897985d-mpnlr                       1/1     Running   0          5m6s    10.244.0.2       k8s-crio-master   <none>           <none>
kube-system   coredns-64897985d-wxmsc                       1/1     Running   0          5m6s    10.244.0.3       k8s-crio-master   <none>           <none>
kube-system   etcd-k8s-crio-master                          1/1     Running   0          5m28s   192.168.122.21   k8s-crio-master   <none>           <none>
kube-system   kube-apiserver-k8s-crio-master                1/1     Running   0          5m24s   192.168.122.21   k8s-crio-master   <none>           <none>
kube-system   kube-controller-manager-k8s-crio-master       1/1     Running   0          5m32s   192.168.122.21   k8s-crio-master   <none>           <none>
kube-system   kube-flannel-ds-4prc6                         1/1     Running   0          73s     192.168.122.22   k8s-crio-worker   <none>           <none>
kube-system   kube-flannel-ds-b64hx                         1/1     Running   0          3m6s    192.168.122.21   k8s-crio-master   <none>           <none>
kube-system   kube-proxy-kv52b                              1/1     Running   0          5m7s    192.168.122.21   k8s-crio-master   <none>           <none>
kube-system   kube-proxy-vg4bt                              1/1     Running   0          74s     192.168.122.22   k8s-crio-worker   <none>           <none>
kube-system   kube-scheduler-k8s-crio-master                1/1     Running   0          5m24s   192.168.122.21   k8s-crio-master   <none>           <none>
shubham@k8s-crio-master:~$
```

Figure 7: All the pods in the Kubernetes cluster and their status nodes with CRI-O runtime

```
shubham@k8s-docker-master:~$ kubectl get nodes -o wide
NAME                STATUS   ROLES                  AGE     VERSION   INTERNAL-IP      EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION      CONTAINER-RUNTIME
k8s-docker-master   Ready    control-plane,master   3m23s   v1.23.6   192.168.122.31   <none>        Ubuntu 20.04.4 LTS   5.4.0-109-generic   docker://20.10.14
k8s-docker-worker   Ready    <none>                 2m7s    v1.23.6   192.168.122.32   <none>        Ubuntu 20.04.4 LTS   5.4.0-109-generic   docker://20.10.14
shubham@k8s-docker-master:~$
```

Figure 8: Kubernetes nodes with Docker runtime

```
shubham@k8s-docker-master:~$ kubectl get pods --all-namespaces -o wide
NAMESPACE     NAME                                          READY   STATUS    RESTARTS   AGE     IP               NODE                NOMINATED NODE   READINESS GATES
kube-system   coredns-64897985d-6v6ph                       1/1     Running   0          2m31s   10.244.0.3       k8s-docker-master   <none>           <none>
kube-system   coredns-64897985d-bgrl2                       1/1     Running   0          2m31s   10.244.0.2       k8s-docker-master   <none>           <none>
kube-system   etcd-k8s-docker-master                        1/1     Running   0          2m44s   192.168.122.31   k8s-docker-master   <none>           <none>
kube-system   kube-apiserver-k8s-docker-master              1/1     Running   0          2m46s   192.168.122.31   k8s-docker-master   <none>           <none>
kube-system   kube-controller-manager-k8s-docker-master     1/1     Running   0          2m47s   192.168.122.31   k8s-docker-master   <none>           <none>
kube-system   kube-flannel-ds-ktcqd                         1/1     Running   0          99s     192.168.122.32   k8s-docker-worker   <none>           <none>
kube-system   kube-flannel-ds-zxwj4                         1/1     Running   0          116s    192.168.122.31   k8s-docker-master   <none>           <none>
kube-system   kube-proxy-4cpht                              1/1     Running   0          99s     192.168.122.32   k8s-docker-worker   <none>           <none>
kube-system   kube-proxy-ggsqt                              1/1     Running   0          2m31s   192.168.122.31   k8s-docker-master   <none>           <none>
kube-system   kube-scheduler-k8s-docker-master              1/1     Running   0          2m52s   192.168.122.31   k8s-docker-master   <none>           <none>
```

Figure 9: All the pods in the Kubernetes cluster and their status nodes with Docker runtime

## CRI-O with Kubernetes

This is a lightweight CRI runtime, tailored specifically for Kubernetes high-level runtime. CRI-O works in coalition with any OCI runtime, to run pods, manage images and pull from any OCI-compatible image registry. We commonly see CRI-O running along with runC and Clear Containers as low-level runtimes.

The installation procedure is quite complex and involves many steps. The complete installation instructions are kept in our GitHub repository, and the link for the same can be found at *https://github.com/shubhamaggarwal890/nginx-vod/blob/master/kubernetes-CRI-o.md*.

Figure 6 shows all the nodes attached to the Kubernetes cluster, and we can see our installed master and worker nodes. The CONTAINER-RUNTIME column shows the installed container runtime along with its version. Figure 7 showcases all the pods attached to the name spaces. We can see certain pods running on the master node and on the worker node under the NODE column.

## Docker Engine with Kubernetes

Docker was originally developed as a monolithic daemon, but it has evolved over time. Its key components are now distributed, and it supports CRI through containerd. In current versions, containerd is installed along with Docker and interacts with CRI. It manages and runs images.

The installation procedure is quite complex and involves many steps. The complete installation instructions are kept in our GitHub repository, and the link for the same can be found at *https://github.com/shubhamaggarwal890/nginx-vod/blob/master/kubernetes-docker.md*.

Figure 8 shows all the nodes attached to the Kubernetes cluster, where we can see our installed master and worker nodes. The CONTAINER-RUNTIME column shows the installed container runtime along with its version. Figure 9 showcases all the pods attached to the name spaces. We can see certain pods running on the master node and on the worker node under the NODE column.

## Containerd with Kubernetes

Containerd is one of the most popular CRI runtimes today. It is designed to fulfill container life cycle and image management, and is considered resource-efficient and more focused than Docker.

The installation procedure is quite complex and involves many steps. The complete installation instructions are kept in our GitHub repository, and the link for the same

```
shubham@k8s-containerd-master:~$ kubectl get nodes -o wide
NAME                    STATUS   ROLES                  AGE     VERSION   INTERNAL-IP      EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION      CONTAINER-RUNTIME
k8s-containerd-master   Ready    control-plane,master   3m17s   v1.23.6   192.168.122.11   <none>        Ubuntu 20.04.4 LTS   5.4.0-109-generic   containerd://1.6.2
k8s-containerd-worker   Ready    <none>                 99s     v1.23.6   192.168.122.12   <none>        Ubuntu 20.04.4 LTS   5.4.0-109-generic   containerd://1.6.2
shubham@k8s-containerd-master:~$
```

Figure 10: Kubernetes nodes with containerd runtime

```
shubham@k8s-containerd-master:~$ kubectl get pods --all-namespaces -o wide
NAMESPACE     NAME                                              READY   STATUS    RESTARTS   AGE     IP               NODE                    NOMINATED NODE   READINESS GA
TES
kube-system   coredns-64897985d-8q544                           1/1     Running   0          2m22s   10.244.0.2       k8s-containerd-master   <none>           <none>
kube-system   coredns-64897985d-j4jb6                           1/1     Running   0          2m22s   10.244.0.3       k8s-containerd-master   <none>           <none>
kube-system   etcd-k8s-containerd-master                        1/1     Running   0          2m43s   192.168.122.11   k8s-containerd-master   <none>           <none>
kube-system   kube-apiserver-k8s-containerd-master              1/1     Running   0          2m49s   192.168.122.11   k8s-containerd-master   <none>           <none>
kube-system   kube-controller-manager-k8s-containerd-master     1/1     Running   0          2m50s   192.168.122.11   k8s-containerd-master   <none>           <none>
kube-system   kube-flannel-ds-jrwlt                             1/1     Running   0          93s     192.168.122.11   k8s-containerd-master   <none>           <none>
kube-system   kube-flannel-ds-lgvz5                             1/1     Running   0          75s     192.168.122.12   k8s-containerd-worker   <none>           <none>
kube-system   kube-proxy-h8tmv                                  1/1     Running   0          75s     192.168.122.12   k8s-containerd-worker   <none>           <none>
kube-system   kube-proxy-qjcmg                                  1/1     Running   0          2m22s   192.168.122.11   k8s-containerd-master   <none>           <none>
kube-system   kube-scheduler-k8s-containerd-master               1/1     Running   0          2m42s   192.168.122.11   k8s-containerd-master   <none>           <none>
shubham@k8s-containerd-master:~$
```

Figure 11: All the pods in the Kubernetes cluster and their status nodes with containerd runtime

can be found at *https://github.com/shubhamaggarwal890/nginx-vod/blob/master/kubernetes-containerd.md*.

Figure 10 shows all the nodes attached to the Kubernetes cluster, where we can see our installed master and worker nodes. The CONTAINER-RUNTIME column shows the installed container runtime along with its version. Figure 11 showcases all the pods attached to the name spaces. We can see certain pods running on the master node and on the worker node under the NODE column.

Kubernetes orchestrates the running containers and helps them scale up and down based on the need of the hour. The application can be easily patched with the latest changes with a simple rollout from Kubernetes, while also handling rollback in case that is needed. The self-healing feature of Kubernetes helps restart the pods if they crash at any time.

The application was successfully installed over the Kubernetes cluster. There is a plethora of ways in which a Kubernetes cluster can be set up. For the purpose of development, an easy installation of minikube is fine, but for a production-based system, one should go for the installation of master and worker nodes. The choice of container runtime lies with the users; they can choose from a wide variety of container runtimes such as containerd, CRI-O, and Docker.

One can easily install the Kubernetes cluster as shown above, but where to install it is the question. Today, various cloud services offer a Kubernetes engine, but none of these solutions are open source. What if some organisation wants to host an in-house solution for the orchestration of containers? Here, Infrastructure as a Service is a move in the right direction, where we can set up a cloud infrastructure that caters to the computation, networking, and storage requirements. With the advent of open source cloud computing infrastructure, an organisation can easily enhance its computing resources in case of traffic growth, giving it an edge over others that are still embracing traditional deployments. END

## References


- Kubernetes (k8s), *https://github.com/kubernetes/kubernetes/*. Last accessed: Feb. 3, 2022.
- Bilgin Ibryam and Burr Sutter. Kubernetes: The evolution of distributed systems, *https://developers.redhat.com/blog/2020/09/23/kubernetes-the-evolution-of-distributed-systems*. Last accessed: Feb. 3, 2022.
- What is Kubernetes? *https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/*. Last accessed: Feb. 3, 2022.
- What is a pod? *https://kubernetes.io/docs/concepts/workloads/pods/. Last accessed:* Feb. 5, 2022.
- 'Container orchestration with Kubernetes' by Vineet Rajula and Prof B. Thangaraju, Open Source For You Magazine, pages 52–55, May 2018.
- Yu-Ju Hong. Introducing container runtime interface (CRI) in Kubernetes, *https://kubernetes.io/blog/2016/12/container-runtime-interface-cri-in-kubernetes/*. Last accessed: Feb. 5, 2022.
- Mugur Marculescu, Introducing GRPC, a new open-source HTTP/2 RPC framework, *https://developers.googleblog.com/2015/02/introducing-grpc-new-open-source-http2.html. Last accessed:* Feb. 5, 2022.
- Minikube start, *https://minikube.sigs.k8s.io/docs/start/*. Last accessed: Feb. 7, 2022.
- Kubeadm, *https://kubernetes.io/docs/reference/setup-tools/kubeadm/*. Last accessed: Feb. 7, 2022.

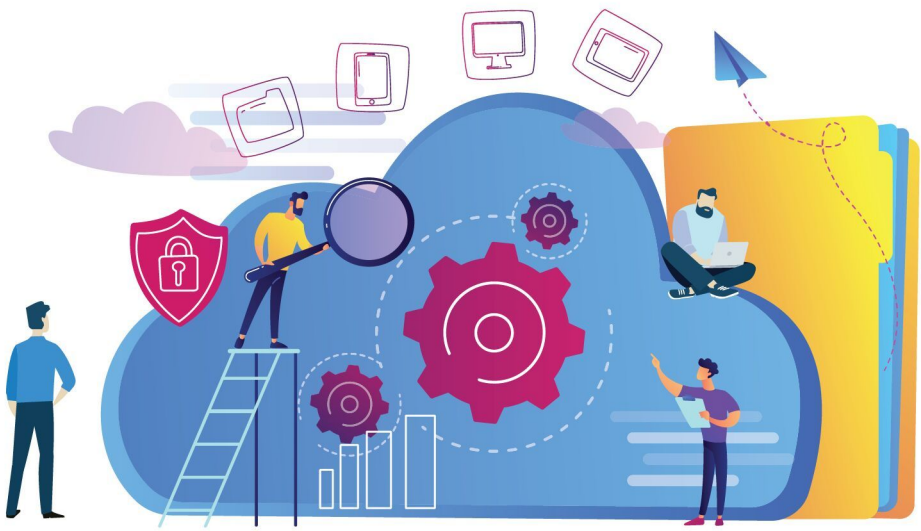


**By: Shubham Aggarwal, Nithya Ganesan, and B. Thangaraju**

The authors are associated with the Open Source Technology Lab at the Internatonal Institute of Information Technology, Bengaluru.

# Cloud Data Management Strategies You Should Adopt

Since the inception of digitalization, data has become a valuable resource for every individual and business. With such humongous data volumes, it has also become necessary to handle them. This article focuses on the different data management strategies that can be followed in the cloud, and tells you why cloud data management is so important.



Various resources help you manage abundant data and guarantee security, which is quite the need of the hour! Previously, enterprises had on-premise data warehouses to keep data and information safe. However, with time, the needs and requirements evolved, leading to an innovative management solution — the cloud!

Let's understand the data management strategy in the cloud in detail!

## What is cloud data management?

Cloud data management helps store an enterprise's data in one or multiple clouds or even on-premises. A highly effective cloud solution provides various features such as data backup, recovery, archiving, analytics, and more. You must know that proper data structuring is essential for an organisation to excel at data management.

## Importance of cloud data management

Cloud data management is beneficial in multiple ways. Here is the list.

1. *Security*: Most enterprises choose cloud data management because of its excellent security. There is no risk of data loss due to hardware failure or system damage. In addition, the service providers ensure implementation of advanced security measures to protect large amounts of data from different companies.
2. *Backups and system recovery*: Most cloud storage providers offer automated data backups to companies to save their time and effort. Backups are an integral part of the management of a company as they help to recover from any damage or ransomware attacks quickly.
3. *Scalability:* Cloud data management allows you to increase or decrease scalability as and when needed for the ever-changing workload.
4. *Automated updates*: Any app receives incremental updates from time to time. Cloud data management providers offer you automated updates. This way, you don't hamper your work regime.
5. *Eco-friendly*: Massive data centres are not eco-friendly. The cloud helps enterprises reduce their carbon footprint and contribute to the environment's well-being.

electronics | embedded | IoT | AI & ML

# India's #1 event for creators of smart solutions based on electronics

# INDIA ELECTRONICS WEEK

## 7-8-9 FEB 2024

KTPO, Whitefield, Bangalore

www.IndiaElectronicsWeek.com

For more information on sponsoring, exhibiting or attending, please call +91-9811155335 or growmybiz@efy.in

## Data management strategy in the cloud

The initial step to creating a data management strategy in the cloud is to understand the requirements of an enterprise well. Every company requires different steps to implement an effective strategy. However, some steps stay constant. Here is a list of some essential strategies.

**Understand your goals:** It is crucial to build data personas to clarify your company's objectives. Data personas represent detailed segments of customers/users in your target audience. Creating these personas helps the workforce to know your consumers more personally, which helps improve conversion rates. You can consider the following data points while making personas for extensive quantitative research:

- Demographics
- Purchasing behaviours
- Affiliations
- Preferences
- Buying patterns

This way, data is managed effectively to produce the most productive results.

Now, as evident, different departments will have different data needs. It should be clear as to who can access them and who cannot! So you must have specific policies for all personas accordingly.

**Implement data governance:** Complying with the data governance policies is a must. All organisations and enterprises must include these policies in the data management strategies in the cloud. As a matter of fact, most companies choose data management on the cloud due to the impeccable safety provisions. Hence, these rules ensure the safety of the data as well. Some of the common compliance frameworks are listed below:

- HIPAA
- GLBA
- PCI-DSS
- FINRA
- SOX, and more

**Five mistakes to avoid with cloud data management**

1. Not making an apt strategy for data protection
2. Not focusing on monitoring cloud performance
3. Not focusing on the risk of cloud sprawling
4. Avoiding role-based access control
5. Ignoring a high-speed internet connection

**Understand which cloud suits you best:** There are different clouds, such as public, private, hybrid, and community clouds, each of which has varied utilisation. It's crucial to understand which one of these clouds will suit your requirements best.

- *Public cloud*: Third parties manage these types of clouds. Public clouds provide reliability, flexibility, high scalability, disaster recovery, and cost reduction. Additionally, public cloud services are a solution for minimal IT infrastructure costs. This option is best for small enterprises that do not have significant investments.
- *Private cloud*: These clouds work on private infrastructure and offer enterprises a certain amount of control over the resources. Private clouds provide high security, compliance with standard procedures, and more. There are various top private cloud providers, such as HP data centres, Microsoft, Ubuntu, etc.
- *Hybrid cloud*: Hybrid clouds or heterogeneous clouds combine the features of public and private clouds. They offer cheap cost, fast speed, and guaranteed safety.
- *Community cloud:* These clouds are a combination of different clouds and tackle the needs of different organisations (industry, business, community, etc). All these organisations share the same infrastructure in a community cloud. Additionally, they are scalable, adaptable, secure, and cost-effective. Generally, the healthcare industry, energy industry, media industry, and scientific organisations use this cloud.

**Maintain authentic data and focus on backups:** Always ensure keeping valid and authentic data. Along with that, it is necessary to track regular backups. If your company hosts its own cloud, make it mandatory for the department to run checks and make a note of the status of backups. Most cloud providers offer automated backups.

**Focus on cloud data management interface (CDMI):** The cloud data management interface, or CDMI, is an international standard that helps provide an interface to manage and access cloud storage. This interface comprises a comprehensive object storage model. Further, the CDMI standard has various security mechanisms to protect data. It uses the Transport Layer Security protocol to provide safety between the client and the CDMI server.

## Wrapping up

You must keep your objectives clear while forming a data management strategy in the cloud. Remember that the process of managing data is ever-evolving, and you must monitor it at regular intervals.

Although there is no rule book to manage cloud data, there are some essential steps that you must take for an effective data management strategy in the cloud. I hope the steps mentioned above help you form a data management strategy that helps you in the best possible way! END

**By: Vijay Singh**

The author is a tech writer and a software developer.

R N I No. DELENG/2012/49440, Mailed on 27/28th of Advance month Delhi Postal Regd. No. DL(S)-01/3443/2022-24
Published on 27th February 2023 Licenced to Post without Pre-Payment Licence No U(S)-56/2022-24
Posted at LPC Delhi RMS Delhi 110006