



Targeted Vulnerability Scanning Powered by **AppCheck**

By Andy Ramgobin
Chief Technology Officer
MergeIT



1. INTRODUCTION

With the attack surface widening and cyber criminals crafting more sophisticated ways of breaching businesses, doing the basics with cyber security has never been more important. Security Posture should be an ever evolving state where regardless of the size of your business, you are practicing posture hardening tactics and consistently developing your security strategy to keep up with the ever evolving landscapes of attacks.

The foundations of a good Security Posture involve some core components that are needed to continue day to day activities as a business knowing that you have done what is possible with the resources and budgets that you have available.

When we talk about the foundations of a good security posture, it starts with core technology areas that are combined with compliance and best practice guidelines. There are plenty of advanced technology areas that can be bolted on but these should be carefully considered as they are expensive and require dedicated security analysts to analyse and correlate the threat information which can be expensive paperweights if not executed correctly. There are so many more areas of security such as DAG (Data Access Governance) & FIM (File Integrity Monitoring) that are considered far more advanced but businesses simply can't deploy their whole roadmap in one year, you need to pick your battles and align with evolving budget plans and cycles.

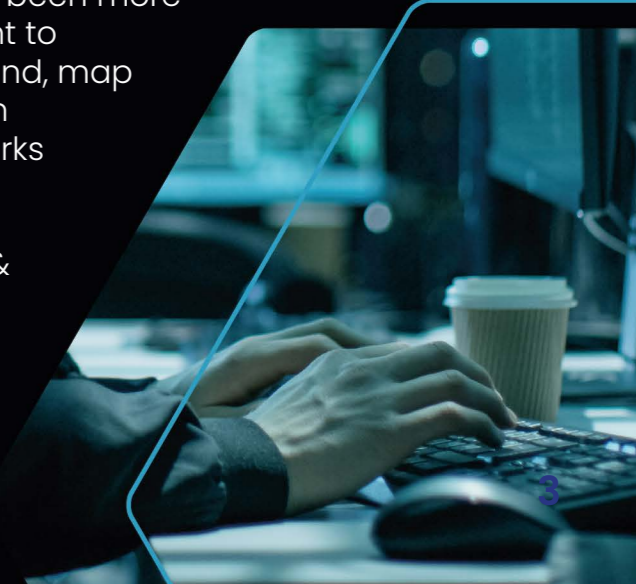
- Vulnerability Scanning & Management
- Penetration Testing
- IAM (Identity Access Management)
- MFA (Multi-factor Authentication)
- PAM (Privilege Access Management)
- EDR (Endpoint Detection & Response)
- NDR (Network Detection & Response)
- NGFW (Next Generation Firewall)
- DLP (Data Loss Prevention)
- NAC (Network Access Control)
- Email Security & Phishing Defense
- Security Awareness Training
- Threat Detection & Monitoring

Everyone wants to talk about Zero Trust and SASE (Secure Access Service Edge) but Zero Trust should effectively start at the Identity, Governance and Federation layer and not just the Network layer. ZTNA (Zero Trust Network Access) in its most simple form is providing a number of authentication mechanisms to authenticate users to different parts of the network and application estate.

With cyber-attacks increasing in their level of sophistication it's never been more important to understand, map and align frameworks such as MITRE ATT&CK &

CONTENTS

1. Introduction	3
2. What is VA Scanning & Why is it Important?	5
3. API Security	9
4. What is API Security?	11
5. OWASP Top 10 - API	11
6. CMS Weaknesses	12
7. The New Face of VA Scanning - AppCheck	13





Lockheed Martin Cyber Kill Chain. Privilege escalation is one of the top TTP's (Tactics, Techniques and Processes) used by cyber criminals.

Vulnerability Scanning & Management tools are an extremely effective deterrent against more typical types of cyber-attack. MITRE ATT&CK is the Adversarial Tactics, Techniques, and Common Knowledge and forms a guideline for classifying and describing cyberattacks and intrusions. TTPs (Tactics, Techniques and Processes) are used to define typical types of attacks and the strategy used by the malicious attack.

There is a particular type of attack where VA Scanning tools are being used by cyber criminals to assess the potential threat surface and then seek to exploit all known attack vectors. Vulnerability scans typically check if the configuration of a target host/application (firmware and version) potentially aligns with the target of a specific exploit the adversary may seek to use.

These scans may also include more broad attempts to gather victim host information that can be used to identify more commonly known, exploitable vulnerabilities. Vulnerability scans typically harvest running software and version numbers via server banners, listening ports, or other network artifacts. Information from these scans may reveal opportunities for other forms of reconnaissance (Search Open Websites/Domains or Search Open Technical Databases), establishing operational resources (develop capabilities or obtain capabilities), and/or initial access (Exploit Public-Facing Application).

Reconnaissance is the first step in the Lockheed Martin Cyber Kill Chain and whilst you can use more sophisticated means to misdirect attackers to a sandboxed, honeypot environment using deception, most businesses are not this far down the roadmap for full stack cyber security. One of the best types of defense in this area is deploy continuous vulnerability scanning

and assessments into your strategy.

A single point of attack can provide a malicious bad actor with an opportunity to move laterally across your estate. If you don't have a mature strategy to protect East and West traffic, you could be leaving yourself open to a much wider and deeper attack on your business.

2. WHAT IS VA SCANNING & WHY IS IT IMPORTANT?

A vulnerability scan is an automated technology that's purpose is to identify vulnerabilities residing in operating systems and third-party software packages using a predefined list of known vulnerabilities. VA scans leverage a knowledge-base of known vulnerabilities (hashes) including those from missing security patches, insecure configuration, potential malware, weak passwords, and more. Commercial offerings include AppCheck, EdgeScan, Qualys, Nessus, Nexpose. Traditional VA scanning tools focus more on internal range scanning which is effectively devices on the network. There are far more vulnerabilities in your environment than just devices on the network.

The quality of these scanners is based on

the depth of their knowledge base and the accuracy of the results (low false positives and false negatives). Opensource scanners make their best effort, however as with a number of Opensource projects, there is a lack of SLA and dedicated support. Combined with restricted funding, Opensource projects often yield mixed results and experiences. Vulnerability scanners are required to be comprehensive, rapidly updated (database), consistent and accurate to ensure their reports are valuable and decisive action can be taken quickly.

OWASP (Open Worldwide Application Security Project) have recently released a Top 10 for API attacks which shows just how attractive an attack vector like APIs have become. GraphQL is a well-known Opensource API library and it is commonly used yet businesses aren't aware of the security vulnerabilities that are associated with REST APIs & Open API. The evolution on SOAP to REST APIs has seen the rapid adoption of API integration and development, however this brings a number of new security risks into your environment.

The explosion of CI/CD (Continuous Integration & Continuous Development) & SDLC (Software Development Lifecycle) in the DevOps arena created the rise of DevSecOps. DevSecOps is a broad term but in essence DevSecOps is a means of integrating security practices within the DevOps process. DevSecOps involves creating a security as code culture with ongoing, flexible collaboration between release engineers, developers and security teams. The DevSecOps movement, like DevOps, is focused on creating new solutions for complex software development processes within an agile framework.

What is 'Shift Left'? 'Shift Left' is the practice of moving security checks as early

and often in the SDLC as possible as part of a DevSecOps shift. Vulnerabilities found earlier in development cycle are much easier and cheaper to fix. A well-known vendor Snyk are the number one vendor in scanning open source code libraries for CWE's (Common Weakness Enumeration), CVE's (Common Vulnerability Enumerations) are well known in the industry and are a glossary that classifies vulnerabilities. The glossary analyses vulnerabilities and then uses the Common Vulnerability Scoring System (CVSS) to evaluate the threat level of a vulnerability. Their sibling CWE's don't always get the headlines. CWE's are a categorised list of software and hardware weaknesses or vulnerabilities.

From a VA Scanning perspective, Dynamic Analysis (DAST) tools, also known as Vulnerability Scanners, such as AppCheck, are typically recommended as the most suitable controls for the "Test" stage of existing pipelines. Many mature SDLC pipelines will typically have an established dynamic testing phase, consisting of automated tools such as Behat or Selenium that provide testing for functional expectations by making HTTP requests to a test or staging deployment of an application. In these scenarios, integrating secure SDLC practices at the Testing stage simply involves extending this phase of testing to include non-functional (security) tests. The building of security into the tools that exist in the DevOps pipeline is typically referred to as "Security as Code (SaC)."

In a Secure SDLC, every build delivered by the development team is immediately scanned in real-time and the developer is notified about issues that they need to correct, these is where SDLC blends into CI – Continuous Integration & CD – Continuous Delivery. CT – Continuous Testing is process of executing automated tests as part of the software delivery pipeline to obtain immediate feedback on the business risks associated with a software release candidate. This information can then be used to determine



if the software is ready to progress through the delivery pipeline at any given time. Micro-services is an evolution of a Service Oriented Architecture or SOA. Micro-services allows you to build application service across multiple Kubernetes clusters which can be in multiple locations and hosted on different architecture platforms. The advantage of leveraging this type of architecture means that a service can be scaled independently by one Kubernetes cluster. That cluster may make up a particular service for the application which needs to be isolated and scaled without affecting another cluster or service. Microservices can be meshed together using a service mesh to deliver full application functionality. Scanning microservices is an essential part of hardening your security posture if you have a mature DevOps approach using CI/CD and SDLC.

There are so many different areas to consider when building a vulnerability management strategy. Typical internal scans is an industry standard however the attack surface is so much wider than this now. External and Web Application scanning is now a requirement if you have public facing infrastructure and applications. The term WAF (Web Application Firewall) has now evolved in WAAP (Web Application API Protection) which goes far beyond the typical DDoS attacks. WAAP's or Next Generation WAF's provide fraud prevention, bot mitigation, payment scraping and RASP (Runtime Application Security Protection). However OWASP Top 10 Web Attacks is now joined by OWASP Top 10 for API Attacks. Whilst Pen Testing is also essential part of a security posture, typically companies only conduct 1x a year towards

the end of the year to meet standards of compliance for auditors. Best practice guidelines are to Pen Test once every 6x months but this isn't commonly adhered to. Breach Attack Simulations should be considered alongside both VA Scanning

alongside both VA Scanning

and Pen testing for a more coherent and complete strategy.

Due to CI / CD, development teams have the ability to reliably release a known secure version of code in weeks, sometimes even days depending on how mature the framework is. To ensure the release is completely secure, the web tier should be

scanned on a regular basis aligning to your release schedule. This is where an enterprise VA Scanning tool, comes into its own, the web tier is still one of the most

attractive attack vectors that cyber criminals will try to exploit.

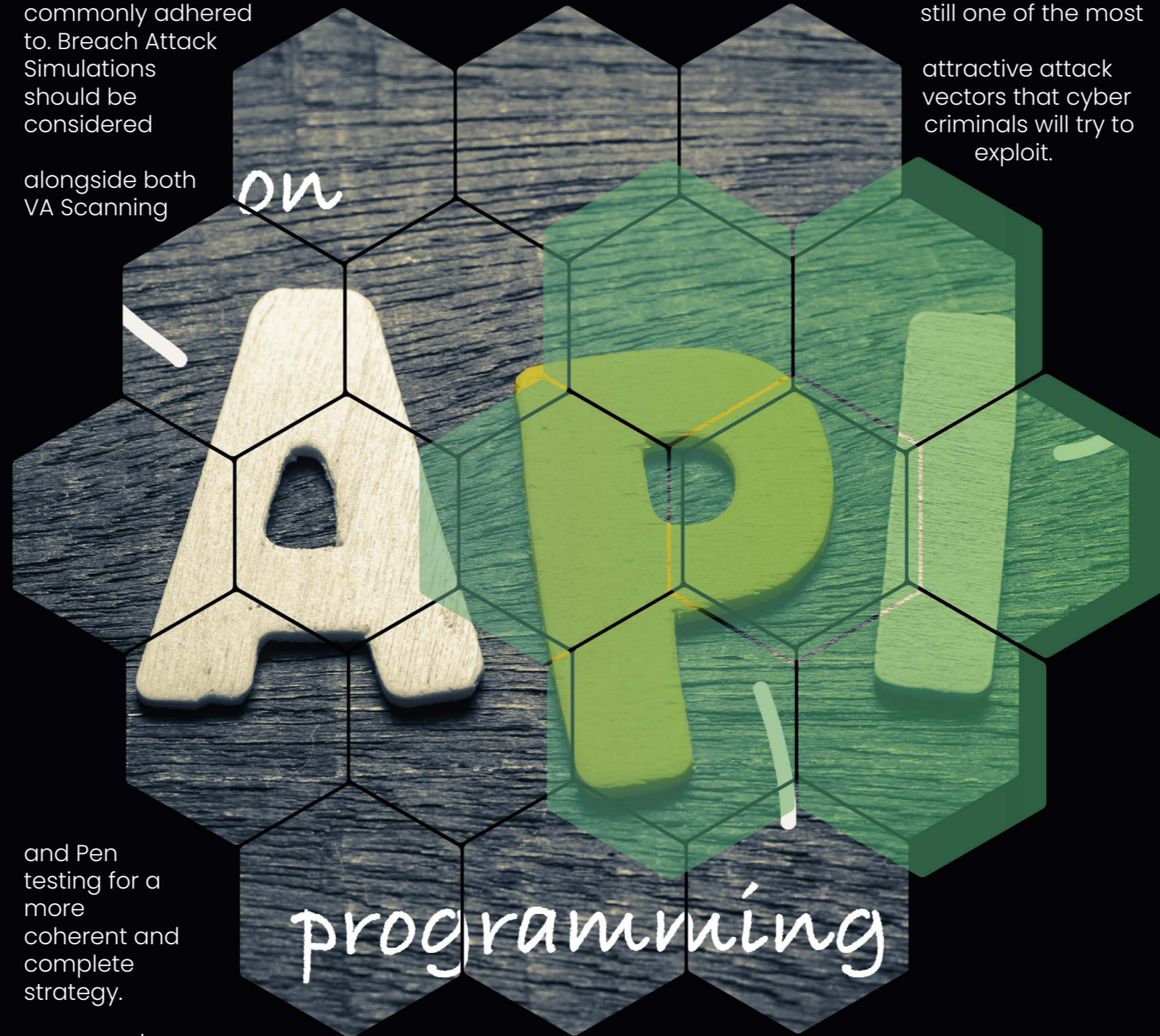
integrating application software.

APIs let your product or service communicate with other products and services without having to know how they're implemented. This can simplify app development, saving time and money. When you're designing new tools and products, or managing existing ones, APIs give you flexibility; simplify design, administration, and use/provide opportunities for innovation.

APIs are sometimes thought of as contracts, with documentation that represents an agreement between parties: If party A sends a remote request structured in a particular way, this is how party B's software will respond.

As web APIs have spread, a protocol specification was developed to help standardize information exchange: Simple Object Access Protocol, more casually known as SOAP. APIs designed with SOAP use XML for their message format and receive requests through HTTP or SMTP. SOAP makes it easier for apps running in different environments or written in different languages to share information.

Another specification is Representational State Transfer (REST). Web APIs that adhere to the REST architectural constraints are called RESTful APIs. REST differs from SOAP in a fundamental way: SOAP is a protocol, whereas REST is an architectural style. This means that there's no official standard for RESTful web APIs. As defined in Roy Fielding's dissertation "Architectural Styles and the Design of Network-based Software Architectures," APIs are RESTful as long as they comply with the 6 guiding constraints of a RESTful system:



3. API SECURITY

What is an API? API stands for Application Programming Interface, which is a set of definitions and protocols for building and

4. WHAT IS API SECURITY?

API security refers to the practice of preventing or mitigating attacks on APIs. APIs work as the backend framework for mobile and web applications. Critical data transfers must be protected from access and exfiltration.

APIs are used in IoT (Internet of Things) applications and on websites. They often gather and process data or allow the user to input information that gets processed within the environment housing the API. For example, there is an API that runs Google Maps. A web developer can embed Google Maps into a page that they are building. When the user uses Google Maps, they are not using code the web designer wrote piece by piece, they are simply using a prewritten API provided by Google. API security covers the APIs you own, as well as the ones you use indirectly. This is an incredibly pertinent area of concern, everyone has the ability to expose their API's for your use, this now creates an even bigger 'Supply Chain like attack' surface which must be protected and mapped.

A foundational element of innovation in today's app-driven world is the API. Containers, Kubernetes, CI/CD & SDLC are intrinsically linked to the explosion in Cloud Native & Application Modernisation. From banks, retail and transportation to IoT, autonomous vehicles and smart cities, APIs are a critical part of modern mobile, SaaS and web applications and can be found in customer-facing, partner-facing and internal applications. By nature, APIs expose application logic and sensitive data such as Personally Identifiable Information (PII) and because of this have increasingly become a target for attackers. Without secure APIs, rapid innovation would be impossible. API

Security focuses on strategies and solutions to understand and mitigate the unique vulnerabilities and security risks of APIs.

Just like 3rd party java script attacks, the daisy chain of API's can provide unique opportunities for hackers to try and exploit. Magecart attacks are defined by payment scraping on the client side. British Airways were fined £20m in 2018 for their Magecart breach and it is essential to take API attacks as seriously as Magecart.

The traditional web application is changing, businesses want native app performance which is driving the adoption of a single page web application. These types of application relies heavily on API calls for resource retrieval.

5. OWASP TOP 10 – API

Every few years the OWASP community come together to review the ten most critical vulnerabilities and security risks relating to Application Programming Interfaces (APIs). This is done by analysing the most important security risks in real-world web APIs, drawing on vulnerabilities and incidents reported by member and partner organisations of OWASP. Identifying the frequency of occurrence as well as impact of common exploit of these vulnerabilities is critical. This list is then published at <https://owasp.org/www-project-api-security/> and is intended to feed into the production of guidance on concerns, countermeasures and best practices relating to security in this area.

The current candidate list for 2023 is:

1. Broken Object Level Authorisation
2. Broken Authentication
3. Broken Object Property Level Authorisation
4. Unrestricted Resource Consumption
5. Broken Function Level Authorisation
6. Server Side Request Forgery
7. Security Misconfiguration
8. Lack of Protection from Automated Threats
9. Improper Assets Management
10. Unsafe Consumption of APIs

OWASP goes into detail on each of these Top 10 attacks using a system to rank their severity, threat agents and attack vectors, security weakness and an explanation of

the blast radius impact. OWASP also discuss whether the API is vulnerable, this information can immediately be collated and fed back into your overall security posture hardening strategy. Not of all of these attacks will apply to every business, but the list should be reviewed every year and steps must be taken to protect businesses from the associated attacks. SME (Small to Medium Enterprises) & Enterprise can no longer ignore or put off designing a specific strategy to secure and protect APIs. VA Scanning natively in the web tier across the full stack is critical moving forward.

6. CMS WEAKNESSES

What is a CMS? A CMS (Content Management System) is a platform which helps in building and delivering web applications quickly so the pipeline

becomes more efficient. Some CMS' are very popular, such as Marketo, WordPress, Drupal, Joomla, and vBulletin. All CMS' require plug-ins and several third-party plug-ins are available for all of these CMSs. It becomes easy to create digital content, handle web content management and enterprise content management.

We have already highlighted the potential risks of plugins and 3rd party tools, this is no different. Every plug-in and CMS is, after all, code. The hackers are intelligent enough to find out the loopholes or bugs in any software system. Thus, they regularly try to attack the CMS, its data and its critical to the business. Consider the points below:

- Widely used CMS' are an extremely attractive attack vector.
- New threat issues and gaps can come up anytime.
- CMS change logs generally show the gaps and vulnerabilities in the versions which are stated in the updates. They also expose the websites which don't update automatically.
- Adding more things to your CMS site increases the attack surface and attractiveness to hackers.

7. THE NEW FACE OF VA SCANNING – APPCHECK

AppCheck was designed from the ground up to emulate the process of a professional Penetration Tester to ensure maximum coverage and accuracy. Rather than use a database of static signatures, AppCheck approaches each test in the

same way a hacker or Penetration Tester would and applies a testing methodology. The vast majority of application security flaws, such as SQL Injection and Cross-Site Scripting, arise from insecure processing of input supplied by the client. AppCheck adopts a first principals approach when testing each input by examining the original expected value and the servers response when the value is modified. By adopting this methodology, AppCheck is able to determine how data may be being processed by the server and can then dynamically evolve each test to identify vulnerabilities. This approach results in more accurate testing and allows AppCheck to identify security flaws that may be masked by security filters and Intrusion Prevention Systems (IPS), but could still be exploited by a real-world attacker.

The AppCheck crawling engine uses a combination of application modelling techniques and subtle heuristic cues to automatically discover the complete attack surface of any given application in the shortest time possible. The algorithms are designed to model how a Penetration Tester or attacker would explore the application, utilising visual cues and ruling out equivalent instances of the attack surface if they have already been explored.

All of this means that for each target discovered, we know its state at discovery and how to recreate that state to later attack it. As the scanner is



AppCheck offers a specific integration with JetBrains TeamCity build management and continuous integration server, as well as an API that can be used to configure, trigger and query scan results from all other major CI/CD pipeline tools.

Ticket Integration:

Once your scan has completed, it's important to know the outcome and to be able to use this for intelligent decisioning. AppCheck provides numerous integration methods including posting of scan outcomes to a webhook, and API polling for scan completion and results delivery. However AppCheck also offers specific integration with the Atlassian JIRA system to deliver findings as tickets into a customer JIRA Cloud ticketing system, to support integration with customer's own ticket system, where they can be assigned for analysis and remediation as needed.

Specific Vulnerability Rescanning:

If you fix one issue, you don't want always to retest your entire application. It is possible to integrate AppCheck with your

issue tracking systems so that issues are automatically retested after being closed by the developer.

Custom Scan Profiles:

AppCheck web application scanner is highly configurable, with scans able to be configured via API or GUI to use one of a number of existing scan profiles, or to define customer-specific scan profiles. If you know that your application uses specific languages and platforms, enable scanning using relevant plugins only, reducing scan times and customising scanning for your specific application for greater scan result accuracy and faster delivery of scan completion and results. This shortens the time to value and allows business to derive value quicker from the data.

Custom Journey Definition:

If a specific code release only changed the code in a specific area of an application, why scan the whole thing? You can save time if your chosen DAST solution enables you to scan only specific portions of your application code that have changed, versus scanning the entire application.

AppCheck supports methods including the restricting of scanning to defined user journeys only. Our custom GoScript allows you to define specific customer journeys to be scanned, delivering similar tightly focused testing to that under systems such as Behat and Selenium and delivering greater scanning efficiency and faster time to delivery.

Infrastructure As Code (IaC) and Web Application support:

Typical DAST solutions don't allow you to scan your platform at both the infrastructure and web application vulnerability layers. Developers are often producing web application code and the operations engineers are deploying artifacts via IaC (Infrastructure As Code), then it makes sense to pick a DAST solution capable of scanning both, automatically, when a deployment of either occurs.

Single Page Application (SPA) Security Scanning:

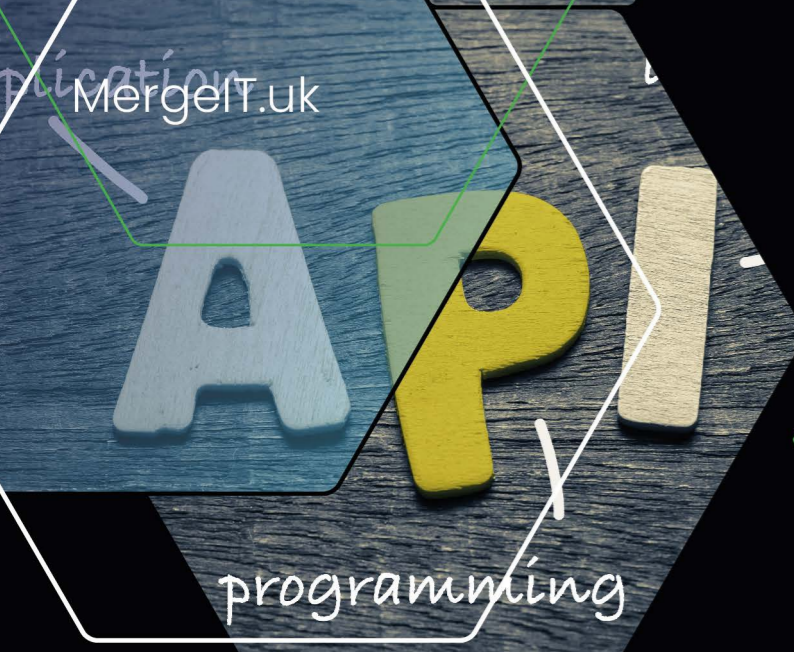
SPAs (Single Page Applications) are a relatively new approach to building web applications which leverage client-side scripting and asynchronous HTTP requests to deliver faster transitions in response to user interactions. The promise of SPAs is the delivery of dynamic web-based applications that mimic the "feel" of native applications that run locally on a device. The popularity of SPAs has soared in recent years as clients familiar with native apps demand the same performance from web applications. The model has been used to power major websites such as Netflix, PayPal, and Google Maps.

behaving in a more human way, it opens up attack vectors that are inaccessible to less sophisticated crawlers.

AppCheck has the ability to deliver Vulnerability Scanning to a variety of different areas, some of them we have already discussed as being critical to hardening your web tier security strategy (API, CMS, Microservices and Internal).

Scan Triggers from CI/CD Pipelines:

A fundamental pre-requisite is that your chosen DAST tool must support integration with your chosen CI/CD pipeline so that you can configure for scans to be triggered automatically once a build is delivered to a test or staging environment.



But new paradigms mean new security challenges. Legacy scanners that depend upon web applications implementing a traditional stateless page-redraw model underperform significantly when tasked with scanning SPAs that leverage rich client-side scripting and make heavy use of API calls for resource retrieval.

With the increased use of API's, DevOps / Cloud Native, microservices & single page applications, a tool like AppCheck can

provide the necessary piece of mind that a vulnerability strategy is hardened and can scale easily as the business scales. It is important to look holistically at security posture and take a pragmatic approach to delivering a hardened strategy. There is no point having SIEM (Security Incident Event Management), SOAR (Security Orchestration Automated Response) or UEBA (User Entity Behavioral Analysis) if you don't respect the fundamental pillars of cyber security.



