

## CSE – 370 CYBER SECURITY

**PROJECT REPORT**

***Submitted by***

Kumaresh N M - E0320004

Dhrish S Kumar - E0320008

Rohit A Ch - E0320022

***In partial fulfilment for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

**(Artificial intelligence and Data Analytics)**

**Sri Ramachandra Faculty of Engineering and Technology**

**Sri Ramachandra Institute of Higher Education and Research, Porur, Chennai -600116**

**APR, 2023**

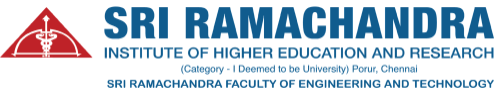
**BONAFIDE CERTIFICATE**

Certified that this project report **“Ethical Hacking”** is the bonafide record of work done by **“Dhrish S Kumar E0320008”** who carried out the work under my supervision.

**Signature of the Supervisor Signature of the Dean**

|  |  |
| --- | --- |
| **Dr. K. Yamini**  **Lecturer,**  Sri Ramachandra Faculty of Engineering and Technology,  SRIHER, Porur, Chennai-600 116. | **Prof. T. Ragunathan**  **Dean,**  Sri Ramachandra Faculty of Engineering and Technology,  SRIHER, Porur, Chennai-600 116. |

**Evaluation Date:**



## ACKNOWLEDGEMENT

I express my sincere gratitude to our Chancellor, Vice-Chancellor and our sincere gratitude to our Provost **Dr.V.Raju** and Our Dean **Prof. T. Ragunathan** for their support and for providing the required facilities for carrying out this study.

I wish to thank my faculty supervisor(s), **Prof.Yamini K** Department of Computer Science and Engineering, Sri Ramachandra faculty of Engineering and Technology, for extending help and encouragement throughout the project. Without his/her continuous guidance and persistent help, this project would not have been a success for me.

I am grateful to all the members of Sri Ramachandra Faculty of Engineering and Technology, my beloved parents and friends for extending the support, who helped us to overcome obstacles in the study.

## INDEX

**S.No Title PageNumber**

[1. Introduction 3](#_Toc27409)

[2. Tools Used 4](#_Toc27410)

[3. Phases Involved 8](#_Toc27411)

[4. ScreenShots 16](#_Toc27413)

[6. Conclusion 41](#_Toc27414)

# 1. INTRODUCTION:

Cybersecurity is an increasingly important field in today's digital age, where the vast majority of business, personal and governmental activities take place online. With the rise of cyberattacks, cybercrime, and other forms of online threats, cybersecurity has become a critical issue that requires attention and resources from individuals, organizations, and governments alike. Cybersecurity tools are essential software applications that help protect computer systems, networks, and sensitive data from cyber attacks, unauthorized access, and other security threats. These tools can help detect and prevent malware, viruses, ransomware, phishing attacks, and other cyber threats.This report will explore the topic of cybersecurity, including its importance, the types of tools and the various measures that can be taken to protect against these threats.

# 2. TOOLS USED:

* **Red Hawk**
* **WireShark**
* **OpenVas**

**Red Hawk:**

Red Hawk is an open-source tool designed for website reconnaissance and vulnerability scanning. It offers various features to gather information about a target website, scan for vulnerabilities, and assess its security. The tool conducts comprehensive information gathering, including obtaining the website's IP address, server details, WHOIS information, geolocation data, and DNS records.One of the main functionalities of Red Hawk is vulnerability scanning. It checks for common vulnerabilities such as SQL injection, cross-site scripting (XSS), local file inclusion (LFI), remote file inclusion (RFI), and more. By identifying these security weaknesses, it helps website owners and security professionals improve the overall security of web applications.

Red Hawk also aids in subdomain discovery, revealing additional attack vectors and providing insights into the site's infrastructure. It can crawl through the target website, extract internal and external links, and analyze HTTP response headers to gather information about the server type, security headers, and cookies. Moreover, the tool offers a WHOIS lookup feature to retrieve registration details and ownership information.In addition, Red Hawk includes a basic port scanning functionality to identify open ports on the target server, which can provide insights into potential services running.

**WireShark**:

Wireshark is a popular open-source network protocol analyzer that allows users to capture and analyze network traffic in real-time. It provides detailed information about network packets, helping users understand and troubleshoot network issues, analyze network performance, and investigate security incidents. Here are some key points about Wireshark:

1. Packet Capture: Wireshark allows users to capture network packets from various interfaces, including Ethernet, Wi-Fi, and USB. It captures packets in promiscuous mode, meaning it captures all traffic on the network segment, including packets not intended for the user's machine.
2. Protocol Analysis: Wireshark supports a wide range of network protocols and can decode and analyze packet contents for protocols such as TCP/IP, HTTP, DNS, FTP, SMTP, and many more. It provides a detailed view of packet headers, payloads, and other protocol-specific information.
3. Filters and Display Options: Wireshark offers powerful filtering capabilities to isolate and display specific packets based on various criteria, such as source/destination IP address, port number, protocol, and packet contents. This helps users focus on the relevant traffic and extract the necessary information.
4. Packet Inspection: Wireshark allows users to inspect individual packets, view their contents, and analyze packet-level details. It provides features like packet coloring, packet marking, and packet annotation, making it easier to identify specific packet types or highlight important packets.

**OpenVas:**

OpenVAS (Open Vulnerability Assessment System) is an open-source vulnerability scanning and management tool. It is designed to identify and assess security vulnerabilities in computer systems, networks, and web applications. Here are some key points about OpenVAS:

1. Vulnerability Scanning: OpenVAS performs comprehensive scans to identify potential vulnerabilities in target systems. It uses a database of known vulnerabilities, checks for misconfigurations, and analyzes network services and applications for security weaknesses.
2. Network Discovery: OpenVAS can discover hosts and devices on the network, providing an overview of the network infrastructure. It helps in identifying all systems that need to be scanned for vulnerabilities.
3. Assessment and Reporting: OpenVAS provides detailed reports on discovered vulnerabilities, including severity levels, affected systems, and recommended remediation actions. These reports help system administrators and security professionals prioritize and address vulnerabilities effectively.
4. CVE Integration: OpenVAS integrates with the Common Vulnerabilities and Exposures (CVE) database, which provides a standardized list of known vulnerabilities. This allows OpenVAS to match identified vulnerabilities with CVE references, enabling accurate identification and classification.
5. User-Friendly Interface: OpenVAS offers a web-based user interface that allows easy configuration, management, and monitoring of vulnerability scans. The interface provides intuitive controls and visualizations for efficient navigation and analysis of scan results

**Acunetix Web Vulnerability Scanner**:

1. Acunetix Web Vulnerability Scanner is a comprehensive web application security testing tool designed to identify vulnerabilities and security flaws in web applications. Here are some key points about Acunetix:
2. Automated Scanning: Acunetix performs automated scans of web applications, analyzing their structure, functionality, and input fields to identify potential vulnerabilities. It scans for common security issues such as SQL injection, cross-site scripting (XSS), and insecure server configurations.
3. Deep Scanning Capabilities: Acunetix employs advanced scanning techniques to thoroughly examine web applications, including crawling and scanning all accessible pages, following links, and scanning multi-step forms. It detects hidden files, directories, and sensitive information that may be exposed.
4. Vulnerability Detection: Acunetix identifies a wide range of vulnerabilities, including injection flaws, security misconfigurations, cross-site request forgery (CSRF), file inclusion vulnerabilities, and more. It provides detailed reports on each identified vulnerability, including severity levels, affected URLs, and recommended remediation actions.
5. OWASP Compliance: Acunetix adheres to the guidelines provided by the Open Web Application Security Project (OWASP), which is a widely recognized authority on web application security. It helps organizations meet the security standards defined by OWASP and ensures compliance with industry best practices.

# 3. PHASES INVOLVED:

### 3.1 Red Hawk

There are five phases involved in this process:

1. Information Gathering
2. Subdomain Discovery
3. Vulnerability Scanning
4. Port Scanning
5. Reporting

Here are the key phases involved when using Red Hawk:

* 1. Information Gathering: Red Hawk starts by gathering essential information about the target website. It collects data such as the website's IP address, server details, WHOIS information, geolocation data, and DNS records. This phase helps in understanding the target's infrastructure and potential attack vectors.
  2. Subdomain Discovery: Red Hawk aids in discovering subdomains associated with the target website. By crawling through the website and analyzing internal and external links, it uncovers additional entry points and provides insights into the target's broader online presence.
  3. Vulnerability Scanning: Red Hawk conducts scanning for common vulnerabilities, including SQL injection, cross-site scripting (XSS), local file inclusion (LFI), remote file inclusion (RFI), and more. It identifies potential security weaknesses in the target website, enabling website owners and security professionals to address these issues.
  4. Port Scanning: Red Hawk includes basic port scanning capabilities to identify open ports on the target server. This information can provide insights into potential services running and help in assessing the overall security posture.
  5. Reporting: Red Hawk generates reports summarizing the findings and results of the reconnaissance and vulnerability scanning phases. These reports provide a consolidated view of the identified vulnerabilities and serve as a basis for further analysis and remediation.

**WireShark**:

1. Planning
2. Setup and Configuration
3. Packet Capture
4. Live Capture Analysis
5. Packet Filtering
6. Packet Inspection and Analysis
7. Statistical Analysis
8. Troubleshooting and Problem Resolution
9. Reporting

. The phases involved in using Wireshark are as follows:

1. Planning: Determine the objective of the network analysis, such as troubleshooting network issues, monitoring network performance, or investigating security incidents. Identify the specific protocols or areas of interest to focus on during the analysis.
2. Setup and Configuration: Install Wireshark on the target machine and ensure it has the necessary privileges to capture network traffic. Configure the network interface to capture packets from the desired network segment. Set any specific capture filters or display filters to narrow down the captured packets.
3. Packet Capture: Start the packet capture process in Wireshark, selecting the appropriate network interface. Wireshark will capture packets flowing through the specified interface and save them for analysis.
4. Live Capture Analysis: Monitor the captured packets in real-time as they are being captured. Analyze packet headers, payloads, and other relevant information to gain insights into network behavior, identify anomalies, or troubleshoot issues.
5. Packet Filtering: Apply filters to the captured packets to focus on specific protocols, source/destination IP addresses, port numbers, or other criteria of interest. Filtering helps in isolating relevant packets and reducing the volume of captured data for analysis.
6. Packet Inspection and Analysis: Analyze individual packets in detail to understand their structure, contents, and interactions. Examine protocol headers, payload data, and any error messages or unusual behavior observed in the packets. Look for patterns, anomalies, or signs of network issues or security threats.
7. Statistical Analysis: Use Wireshark's built-in statistical features to analyze network performance metrics, such as packet rates, throughput, response times, or error rates. Generate graphs or statistics to visualize trends or identify performance bottlenecks.
8. Troubleshooting and Problem Resolution: Based on the analysis findings, identify network issues or security vulnerabilities. Take appropriate actions to resolve the identified problems, such as adjusting network configurations, addressing software or hardware issues, or implementing security measures.
9. Reporting: Document the analysis findings, including any identified problems, recommended solutions, or noteworthy observations. Prepare a comprehensive report that can be shared with relevant stakeholders or used for future reference.

**OpenVas**:

* Configuration
* Target Selection
* Scan Configuration
* Scan Execution
* Vulnerability Assessment
* Vulnerability Reporting
* Remediation Planning
* Remediation Execution
* Continuous Monitoring

OpenVAS follows a series of phases to conduct comprehensive vulnerability assessments:

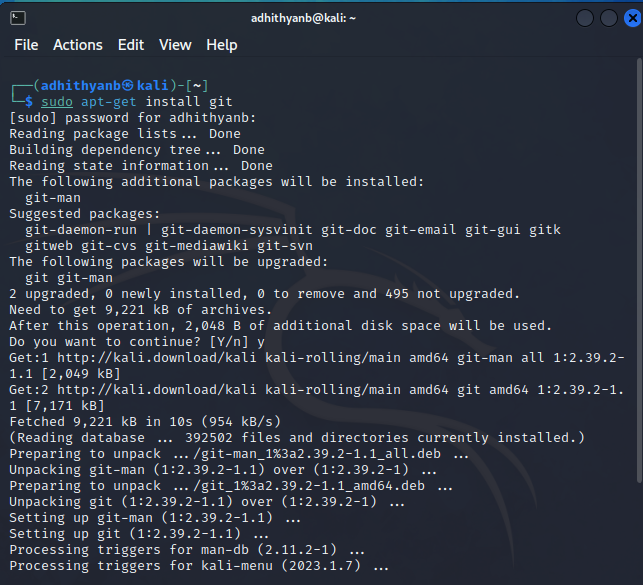
1. Configuration: Set up the OpenVAS environment by installing and configuring the necessary components, such as the OpenVAS Scanner, OpenVAS Manager, and OpenVAS Greenbone Security Assistant (GSA). Configure user accounts, permissions, and scan targets.
2. Target Selection: Define the target systems or networks that will be scanned for vulnerabilities. Specify the IP addresses or domain names of the target assets to be assessed.
3. Scan Configuration: Configure the specific parameters for the vulnerability scan, such as the scan intensity level, scan timing options, and selection of vulnerability tests. Customize the scan according to the specific requirements and desired depth of analysis.
4. Scan Execution: Initiate the vulnerability scan by running the OpenVAS scanner. The scanner sends various probes and tests to the target systems to identify vulnerabilities, misconfigurations, or potential security issues.
5. Vulnerability Assessment: OpenVAS analyzes the scan results and identifies vulnerabilities based on known security issues and weaknesses. It assesses the severity and potential impact of each vulnerability and provides detailed information about the vulnerabilities found.
6. Vulnerability Reporting: OpenVAS generates comprehensive reports containing detailed information about the identified vulnerabilities, including their severity, description, and recommended actions for remediation. The reports can be customized and exported in various formats for further analysis and sharing with stakeholders.
7. Remediation Planning: Based on the scan results and vulnerability reports, develop a remediation plan to address the identified vulnerabilities. Prioritize the vulnerabilities based on their severity and potential impact on the systems and networks. Determine the necessary actions to mitigate or eliminate the vulnerabilities.
8. Remediation Execution: Implement the remediation actions according to the planned approach. Apply security patches, update software versions, reconfigure systems, or implement other security measures to address the identified vulnerabilities.
9. Continuous Monitoring: Establish a regular vulnerability management process using OpenVAS to continuously monitor and assess the security posture of the systems and networks. Conduct periodic vulnerability scans, analyze the results, and take appropriate actions to maintain a secure environment.Top of FormBottom of Form

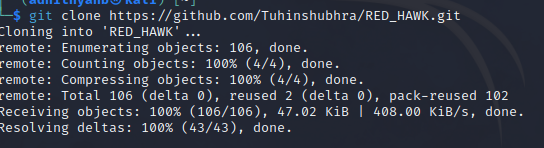
**Acunetix Web Vulnerability**:

The phases involved in Acunetix Web Vulnerability Scanner are as follows:

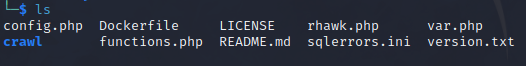
1. Target Identification
2. Crawling
3. Scanning
4. Vulnerability Detection
5. Reporting
6. Remediation
7. Continuous Monitoring
8. Target Identification: In this phase, the target web application is identified and added to the scanning scope. The user provides the URL or IP address of the application to be scanned.
9. Crawling: Acunetix scans the target web application by crawling through its structure, discovering all accessible pages, and following links. This phase helps in mapping the application's architecture and identifying the scope of the scan.
10. Scanning: Once the crawling phase is complete, Acunetix starts scanning each discovered page for vulnerabilities. It performs a variety of security tests and checks for common vulnerabilities such as SQL injection, cross-site scripting (XSS), insecure server configurations, and more. Acunetix utilizes both black-box and white-box testing techniques to thoroughly analyze the application's security posture.
11. Vulnerability Detection: During the scanning phase, Acunetix identifies and categorizes the vulnerabilities it discovers. It assigns severity levels to each vulnerability based on its potential impact and likelihood of exploitation. The scanner generates detailed reports for each identified vulnerability, including information about the affected URL, the vulnerability type, and recommended remediation steps.
12. Reporting: Acunetix generates comprehensive reports that provide a clear overview of the identified vulnerabilities, their severity, and associated details. The reports can be customized to suit specific requirements and exported in various formats, such as PDF or HTML. These reports help security teams and developers prioritize and address vulnerabilities effectively.
13. Remediation: After receiving the vulnerability reports, the development and security teams can collaborate to remediate the identified issues. Acunetix provides guidance on how to fix each vulnerability and offers recommendations to mitigate the risk. The remediation phase involves fixing the vulnerabilities and retesting the application to ensure the effectiveness of the applied patches.
14. Continuous Monitoring: Acunetix supports continuous monitoring and scheduled scans to ensure ongoing security of the web application. Regular scans help identify new vulnerabilities introduced by application updates or emerging threats. Continuous monitoring allows organizations to maintain an updated security posture and respond to potential risks in a timely manner.

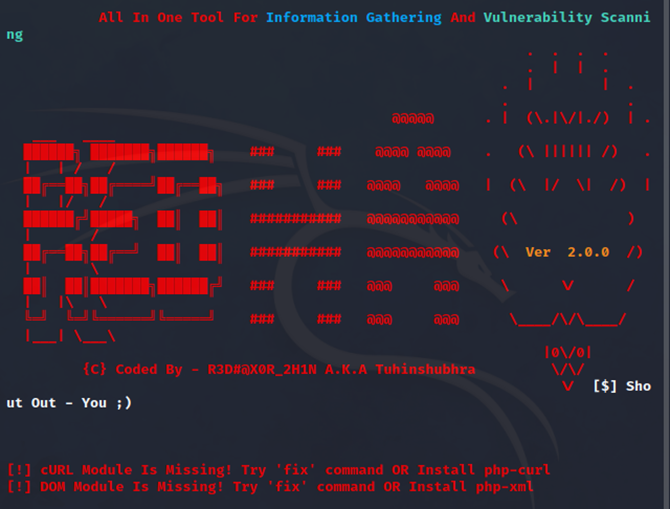
**Red Hawk:**



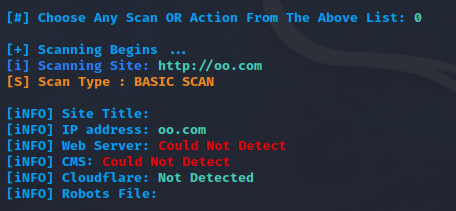




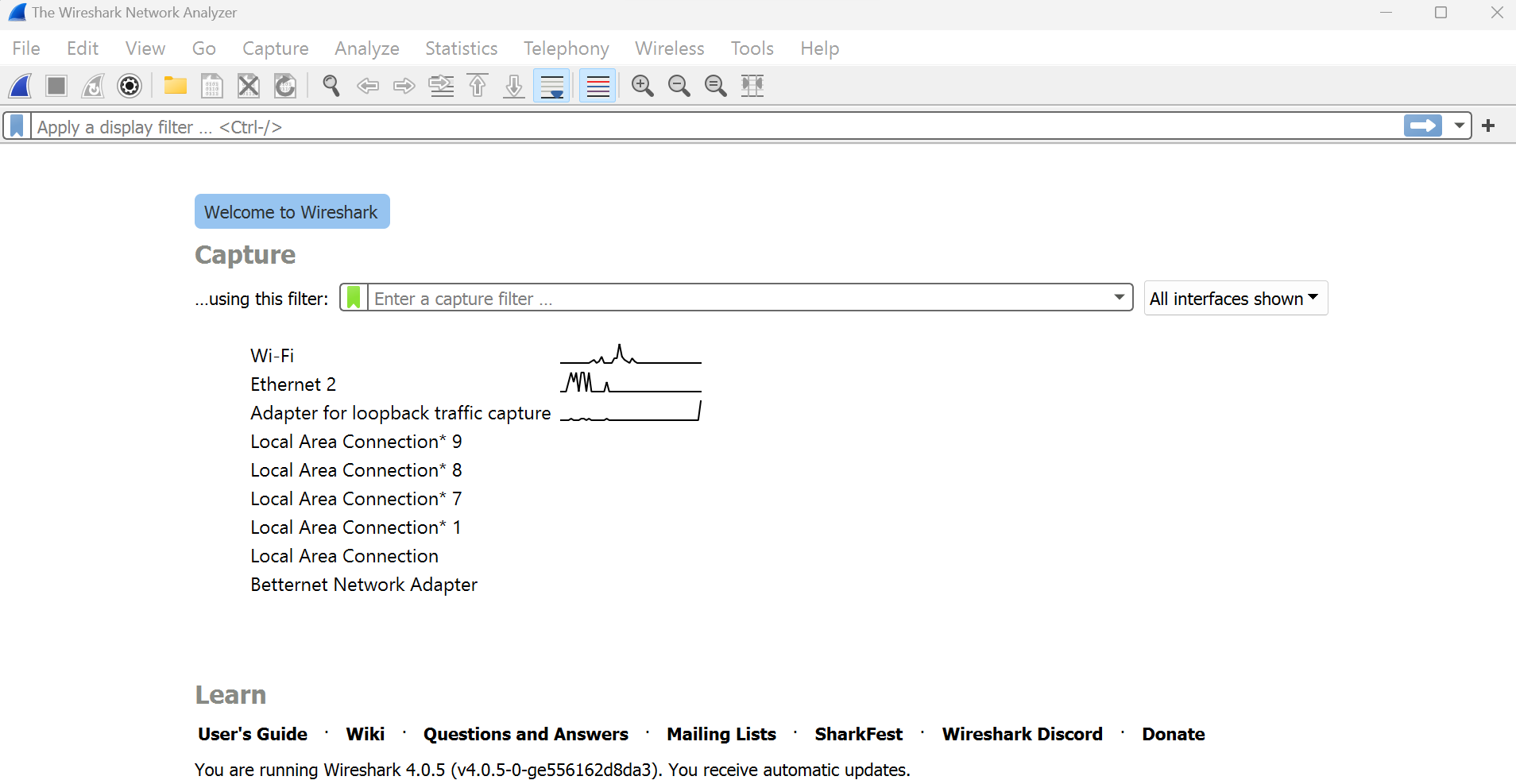


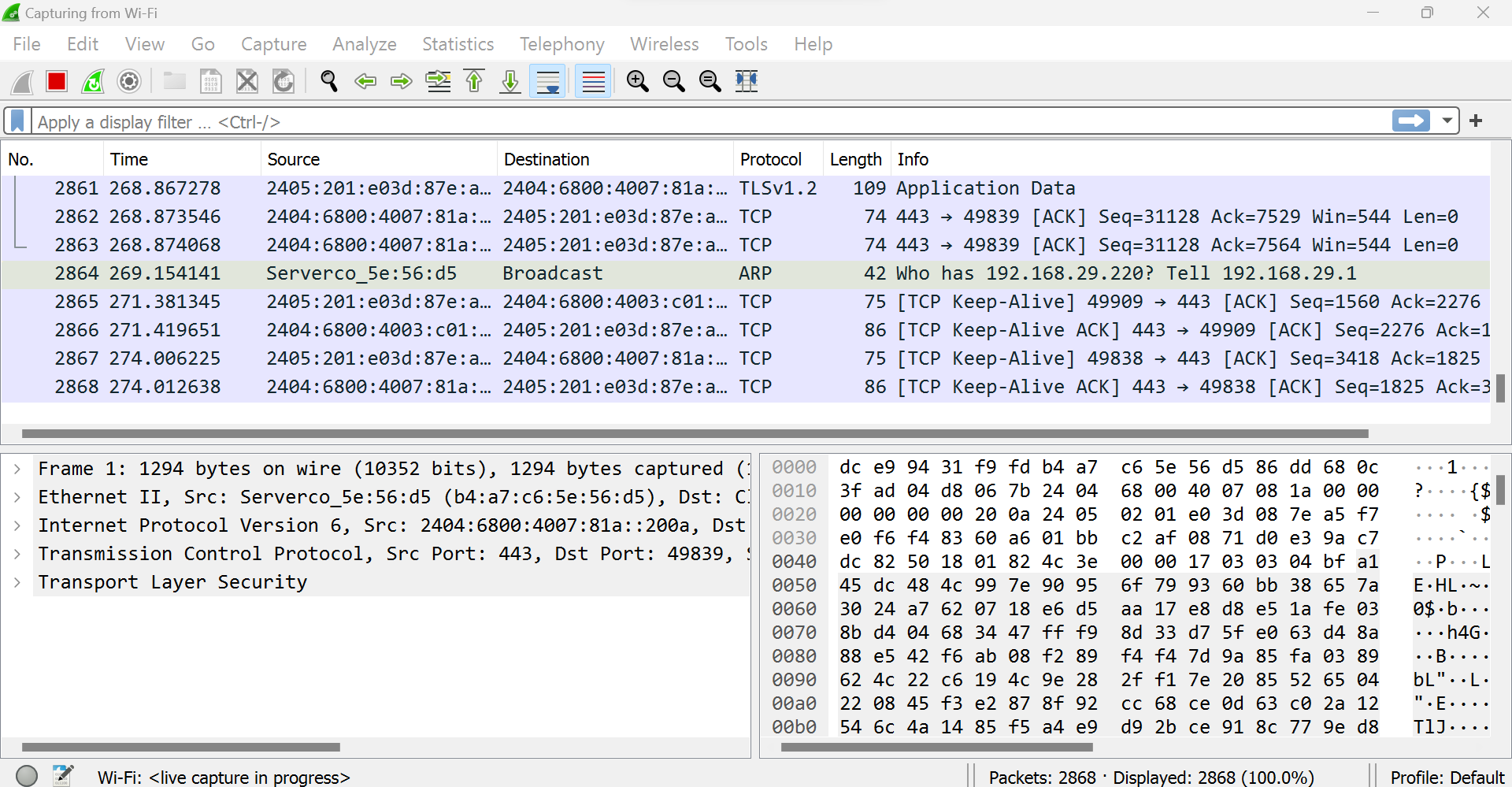


# 

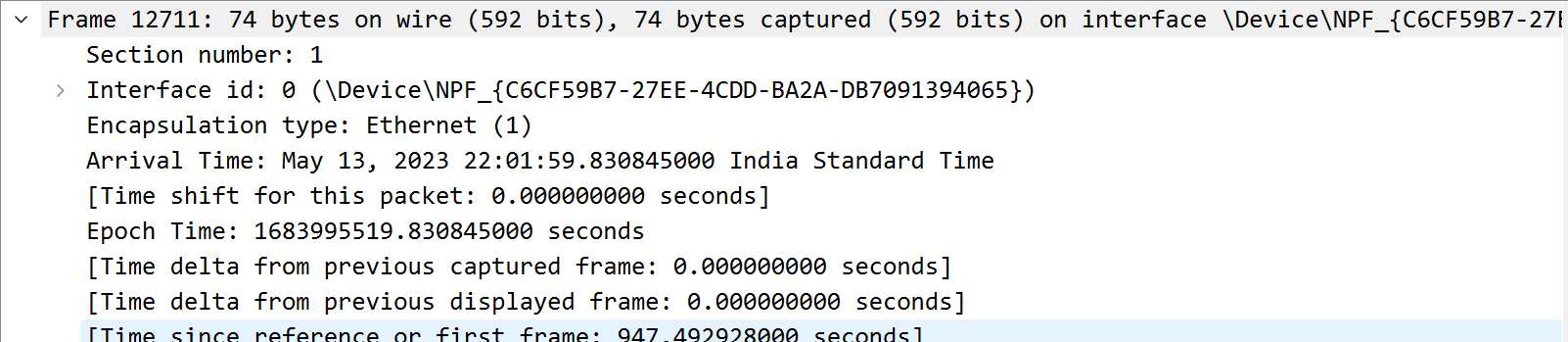


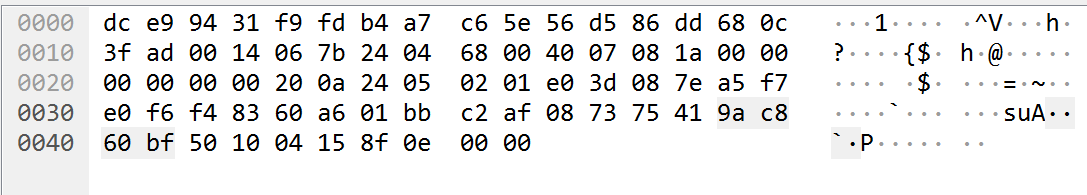
# WireShark:

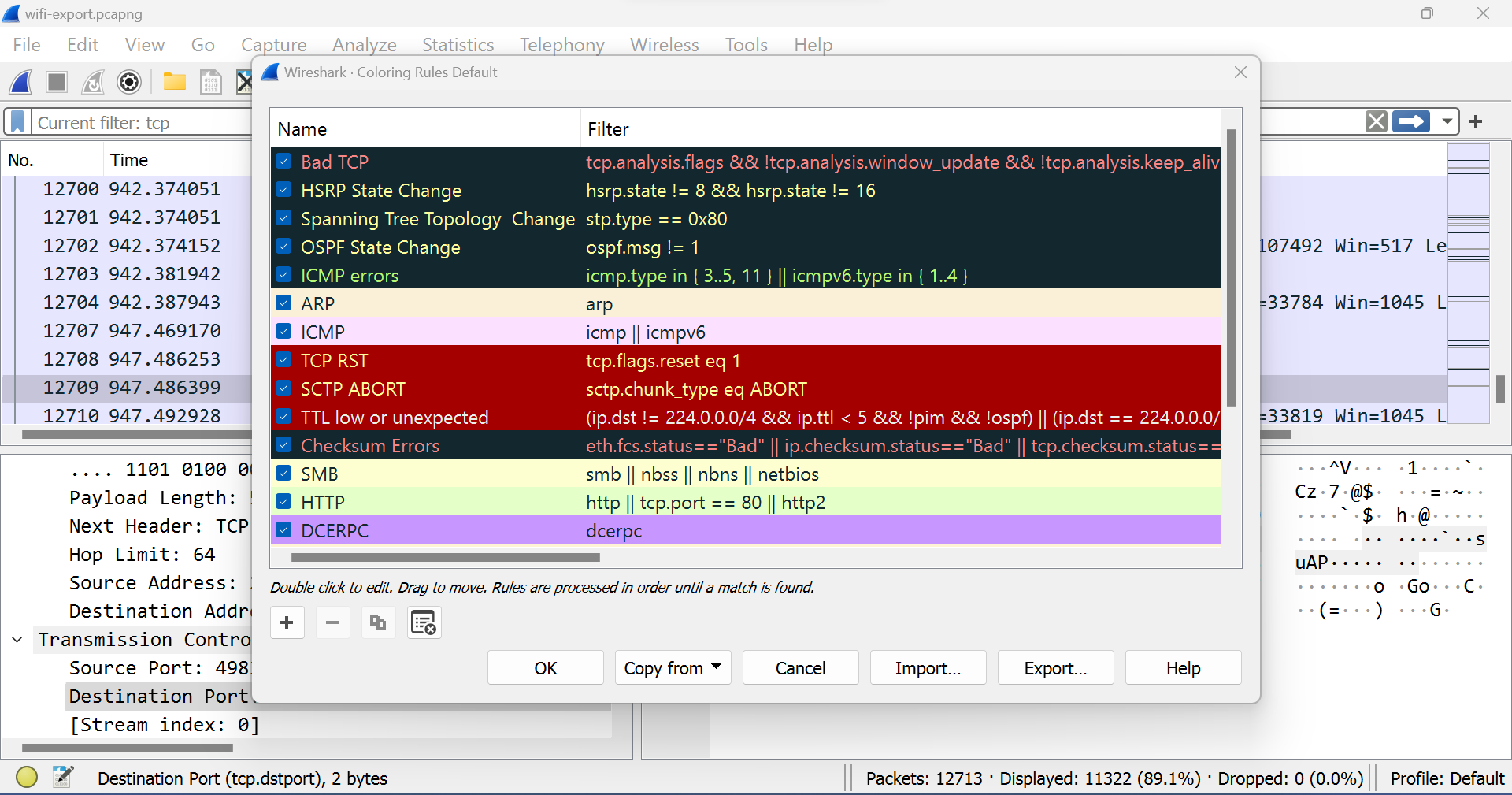




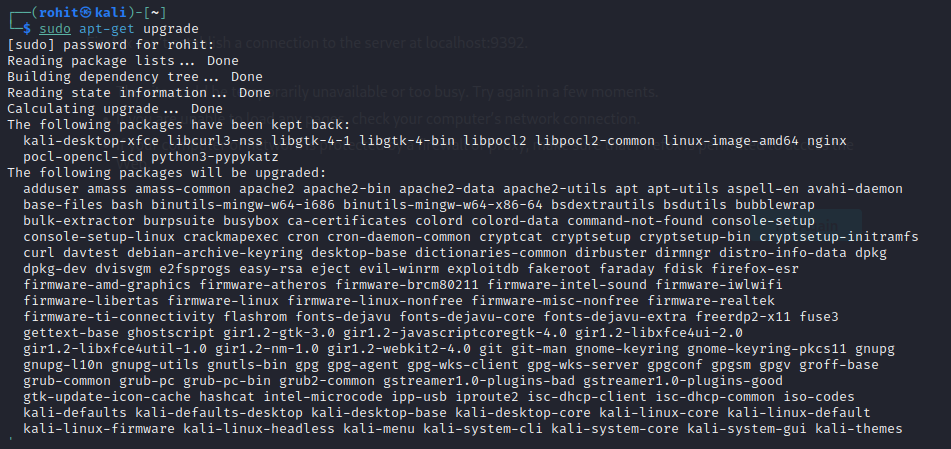
# 

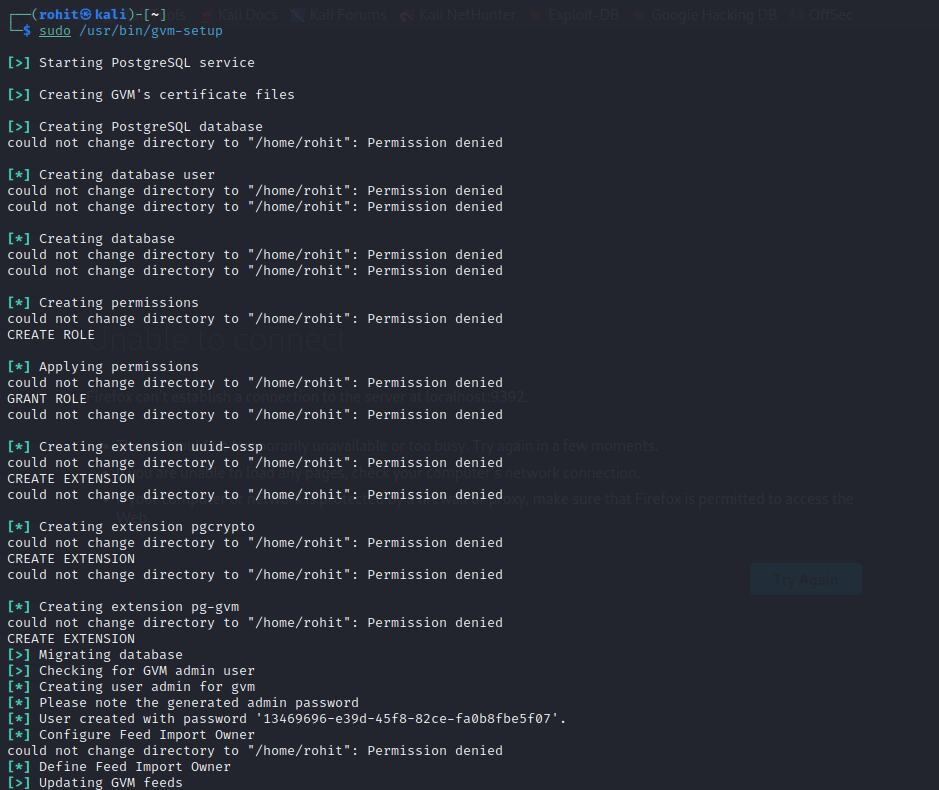




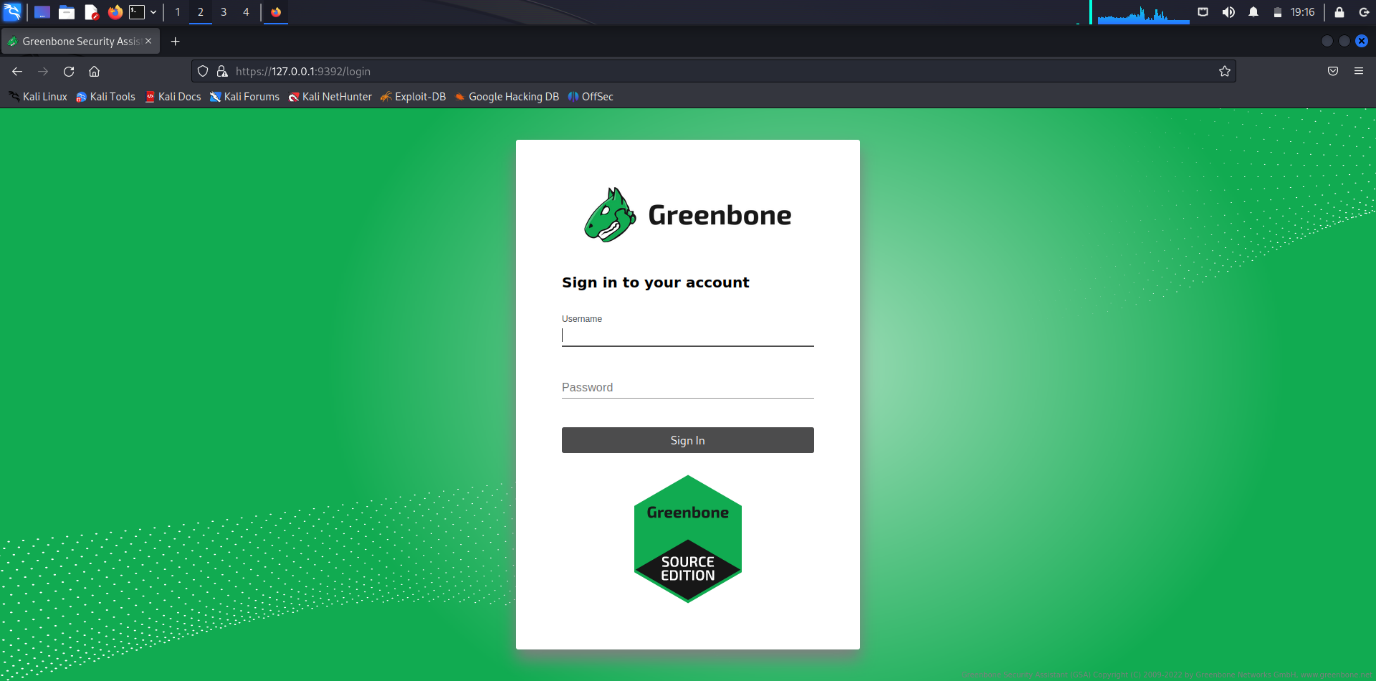


# OpenVAS:





# 



# 4. Acunetix Web Vulnerability Scanner

**4.1 Working and Output:**

### Performing an Online Vulnerability Scan

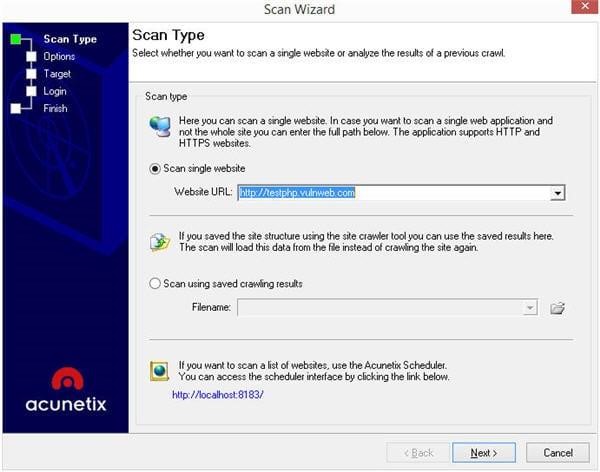
Before starting a scan, I needed a vulnerable site to test. Acunetix maintains its own test sites which you can scan to test the product.

* http://testhtml5.vulnweb.com
* http://testphp.vulnweb.com
* http://testaspnet.vulnweb.com
* http://testasp.vulnweb.com

Starting a new scan is as simple as starting the *Scan Wizard* by clicking the *New Scan* button in the main toolbar. The wizard will walk you through some options you can use to customize the scan.

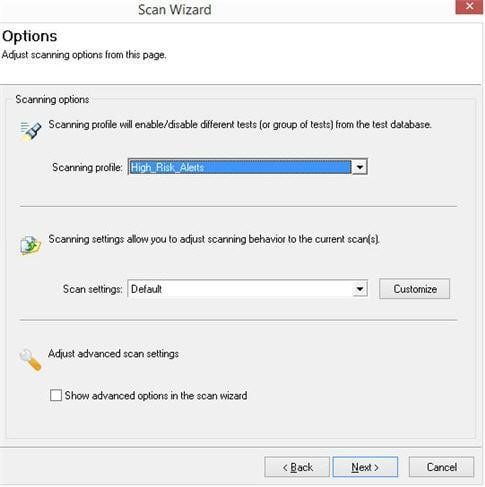


We first need to tell Acunetix Web Vulnerability Scanner what site we’d like to scan. In this case, I’ll be sticking with the PHP test site above (i.e. http://testphp.vulnweb.com).

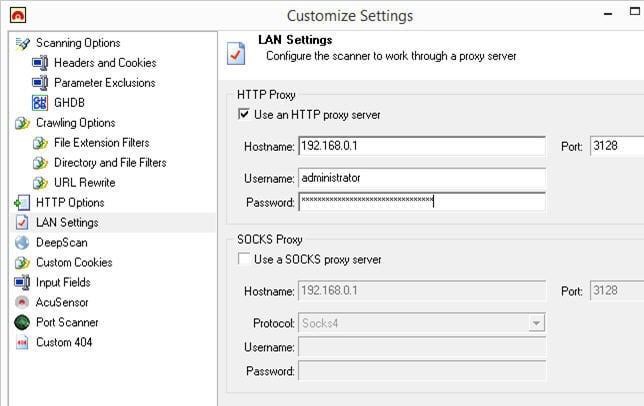


Next, we’ll need to select a *Scanning Profile*. A Scanning Profile is a logical grouping of tests that perform a specific group of tests. This feature allows you to customize what tests you want or don’t want Acunetix WVS to run. You can choose from the several built-in Scanning Profiles, or you can create custom Scanning Profiles that suit your specific requirements.

The *Default* Scanning Profile includes every test Acunetix Web Vulnerability Scanner can run. However, let’s assume I’m only concerned about high-risk alerts, I can customize the scan to the only test for those vulnerabilities.

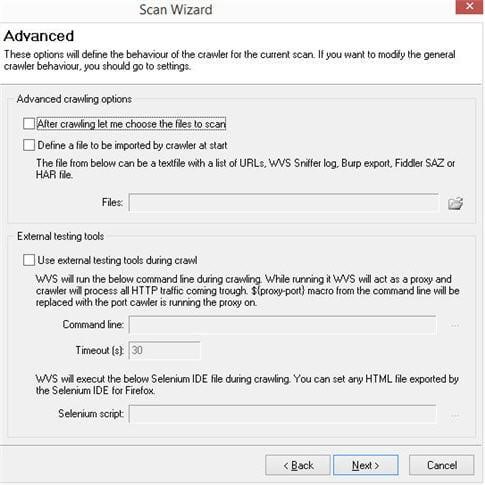


Scanning Profiles are not the only way to customize a scan — *Scan Setting* allows very granular control over your scan. Most users will not need to modify these settings since the defaults have been carefully selected to cater for the vast majority of websites and web applications. However, since I happen to be connecting to the internet using an HTTP proxy, I’ll go ahead and configure that from here by clicking the *Customize* button next to the Scan Settings list box.



Should you need them, Acunetix WVS also has advanced options you can leverage if you need even more control over the pages you want (or don’t want) the scanner crawl and scan.

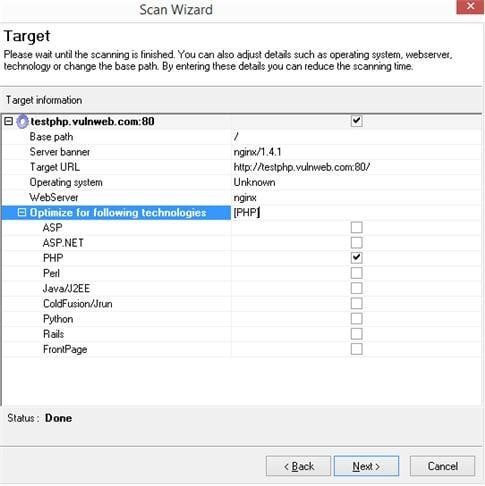
You can select which pages you want to exclude from a scan using the *After crawling let me choose the files to scan* option, and even import results from other tools such as Portswigger’s BurpSuite and Telerik’s Fiddler, and of course Acunetix WVS’ built-in HTTP Sniffer.



Being a black-box scanner, Acunetix WVS can scan any website or web application, regardless of the technologies, or programming languages it uses — it essentially tests a website or web application without any prior knowledge of how that site works, just like a real attacker would.

**Scan Optimization:**

Having said this, Acunetix Web Vulnerability Scanner has some intelligent tricks up its sleeve to optimize the scan for a specific technology. Acunetix WVS will try to fingerprint the web application in order to detect the technologies it is using to cut-down on the scan time. E.g. If I’m testing a site built using PHP, there is no reason to look for vulnerabilities that can only exist in ASP.NET applications.



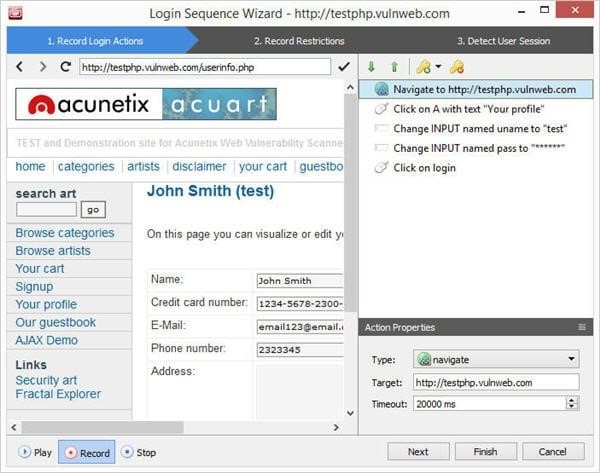
### How to Scan Password Protected Areas of a Website

Because this site has a login page, we need to create a *Login Sequence* in order to instruct the scanner on how to log into the application. This is an essential part of the scanning process and something that is usually difficult or tedious to set-up properly with other scanners.

You can either attempt to have the scanner log in for you (this will work for most simple sites with just a username and password), or else you can create a Login Sequence manually (works better for more complex logins and provides much more control).



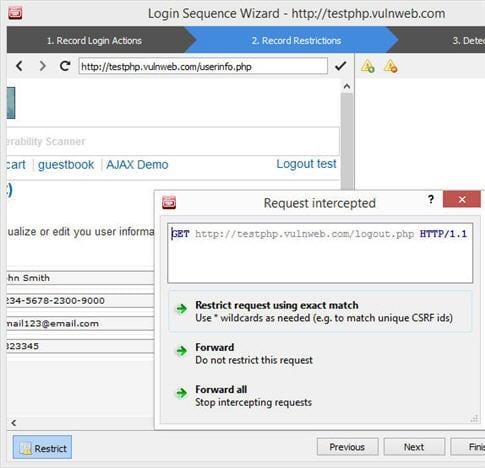
Acunetix Web Vulnerability Scanner makes creating a Login Sequence dead-easy, simply go through your normal login process of signing into an account; you’ll notice that your actions are being recorded. The scanner will replay these actions to log in during the scan.



You can also use the replay button at the bottom-left of the *Login Sequence Recorder* window to replay your actions just to make sure everything is working correctly.

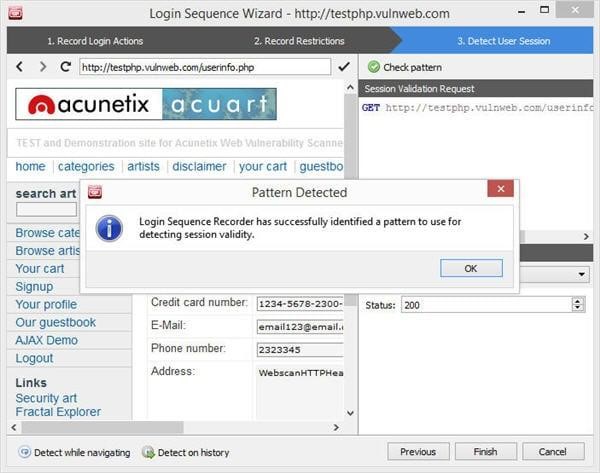
Once you click *Next* you have the option of selecting what links you do not want the scanner to click on while logged in. We obviously don’t want the scanner to get logged out of the session during a crawl or a scan, so I’ll be clicking on the *Logout* link in order to restrict it, however, you are free to set-up as many restrictions as you like.

It’s also worth noting that the Login Sequence Recorder also has support for restricting links with *nonces* (one-time tokens in links) by using wildcards.



Once you’re done restricting links, click *Next*. A Login Sequence alone is not enough. The scanner needs to understand when it is logged in and when it is logged out. The Login Sequence Recorder needs what is known as a *Session Pattern*.

A Session Pattern is nothing more than something unique between a logged in and a logged-out state of a web application. The Login Sequence Recorder will detect this pattern automatically for you; however, you’re free to customize this pattern if you wish to do so.



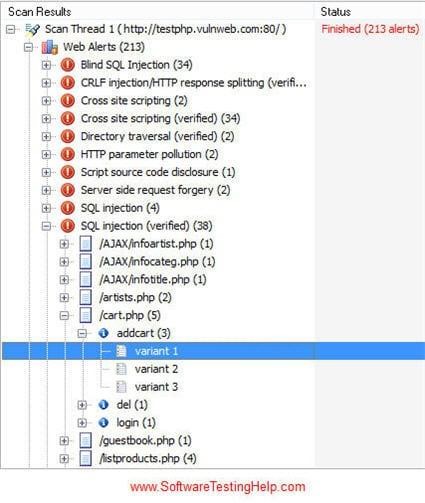
Clicking *Finish* will ask you to save the Login Sequence you’ve just created. This can be used at a later date so you don’t need to go through the process of creating a Login Sequence every time you want to scan the same site.

You will then be presented with the final screen of the Scan Wizard which gives you the option of saving any Scan Settings you might have set. In addition, Acunetix WVS is smart enough to identify if a site provides a different response to a mobile User-Agent string and it will ask you if you’d like to change your User Agent string to say that of an iPhone or an Android device — handy if your site is mobile friendly.

**Website Vulnerability Scan Results:**

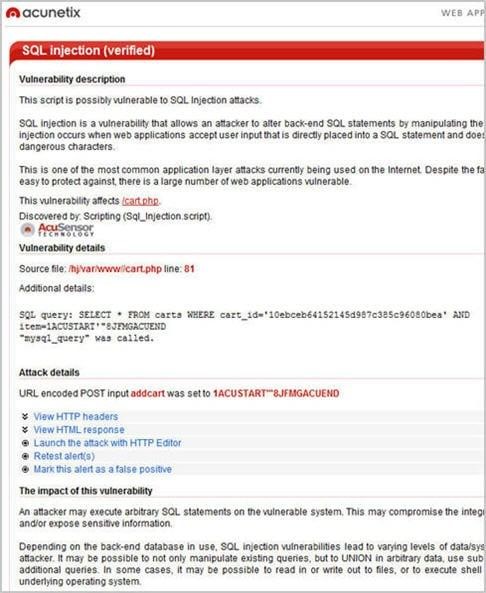
After the crawl and scan are complete, Acunetix WVS will list a list of high-severity vulnerabilities that it detected on the test site.

The moment you click on a specific vulnerability (SQL Injection in this case), Acunetix WVS reveals not only which input parameter is vulnerable but it will also list variations of an attack on that parameter.



Selecting one of the variations of vulnerability explains the vulnerability in great detail. The scanner will first provide a summary of the vulnerability, and then it will proceed to explain what the impact of such vulnerability is and how to fix the vulnerability.

If you’ve installed Acunetix AcuSensor (this is optional), a server-side component for PHP and .NET applications that communicate with Acunetix WVS results for vulnerabilities such as SQL Injection will even include the file and the vulnerable line of code!

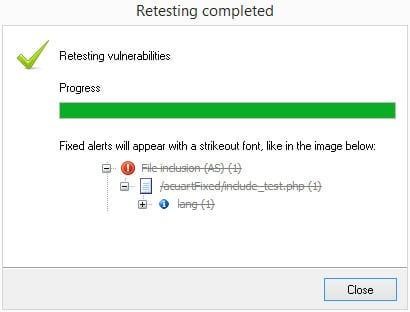


The alert will then provide you with further information containing a lengthier explanation of the problem, as well as more details on how to fix the vulnerability together with a list of reference URLs where you can read up more about the subject, just in case the scanner found something you’re not quite familiar with.

### Re-running Tests After the Vulnerability Fix

Re-running the scan from the start is obviously one way of checking if the fix for a detected vulnerability is successful. However, Acunetix WVS has a very handy *Retest* feature.

Simply right-click an alert you’d like to retest and select *Retest alert(s)*. The tests that detected that vulnerability will be re-run and the new result will be shown. If the vulnerability is resolved, Acunetix will mark it in a gray, strike-through font.



### Web Vulnerability Scan Reporting

From here you can save the scan’s results or generate a variety of easy to understand reports. You can generate reports by clicking the *Reporter* button in the main toolbar.



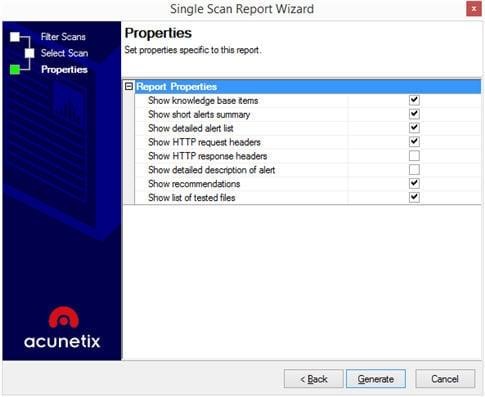
When the Acunetix Web Vulnerability Scanner Reporter loads, you’re presented with a selection of reports you can choose from. If you’re after high-level reports, the *Affected Items*, *Executive Summary*, and *Quick Report* provide a variety of concise reports to choose from.



If on the other hand, you’re after compliance reports, the Acunetix reporter can generate reports tailored to a compliance standard of your choice, be that the OWASP Top 10, PCI, HIPPA or any of the other Compliance Reports available. These reports are periodically updated to always be in-line with the latest version of a compliance standard.

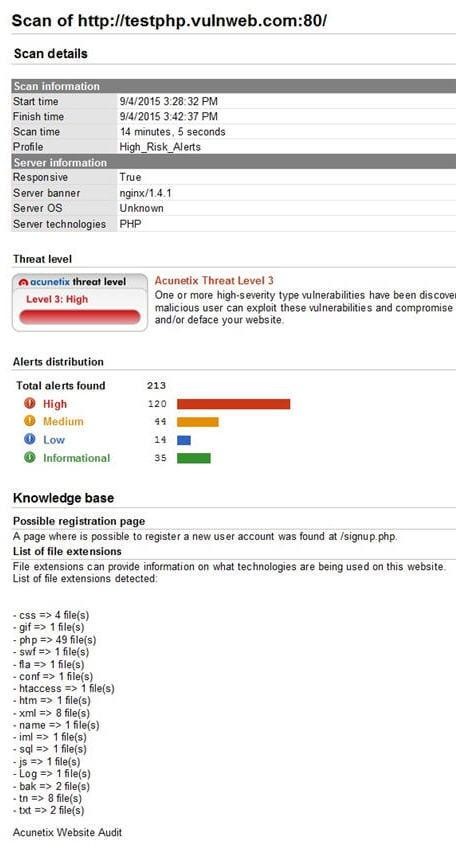


The most detailed report is the *Developer Report*. This report is also highly configurable, allowing the user to include just the necessary information in the report.



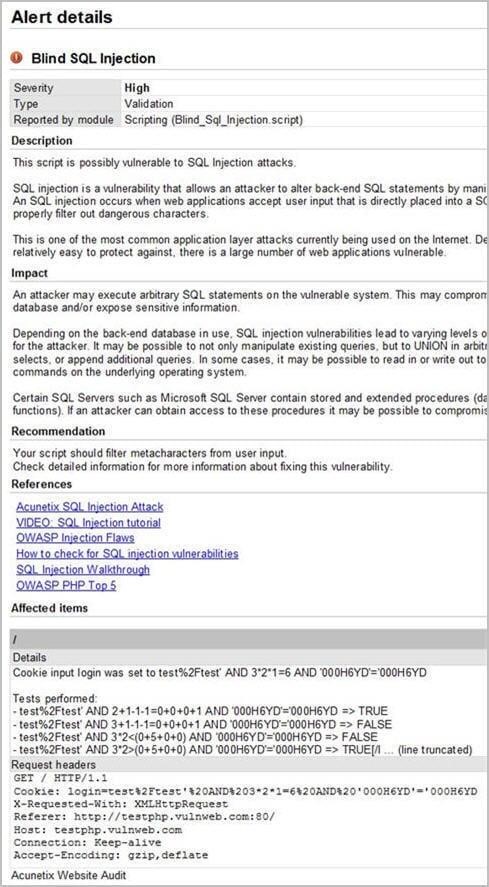
Clicking *Generate* will produce a report which you can save out to PDF, HTML, and other formats to share with colleagues and other stakeholders.

**Summary Page:**





**Alert Details:**



# 6. Conclusion :

In conclusion, cybersecurity tools are essential in protecting computer systems, networks, and sensitive information from various cyber threats. There are many types of cybersecurity tools available, each with their own unique set of features and benefits. These tools can help identify and mitigate security risks, detect and respond to cyber attacks, and provide visibility and control over network activity. Some popular cybersecurity tools include firewalls, antivirus software, intrusion detection and prevention systems, data loss prevention tools, and security information and event management (SIEM) systems. These tools can be used individually or in combination to provide comprehensive cybersecurity coverage. Overall, cybersecurity tools are a critical component of any organization's cybersecurity strategy and should be used in conjunction with best practices and policies to create a comprehensive approach to cybersecurity.