JASON DION
TRAINING THE CYBER SECURITY WORKFORCE

Web App Vulnerability Scanning

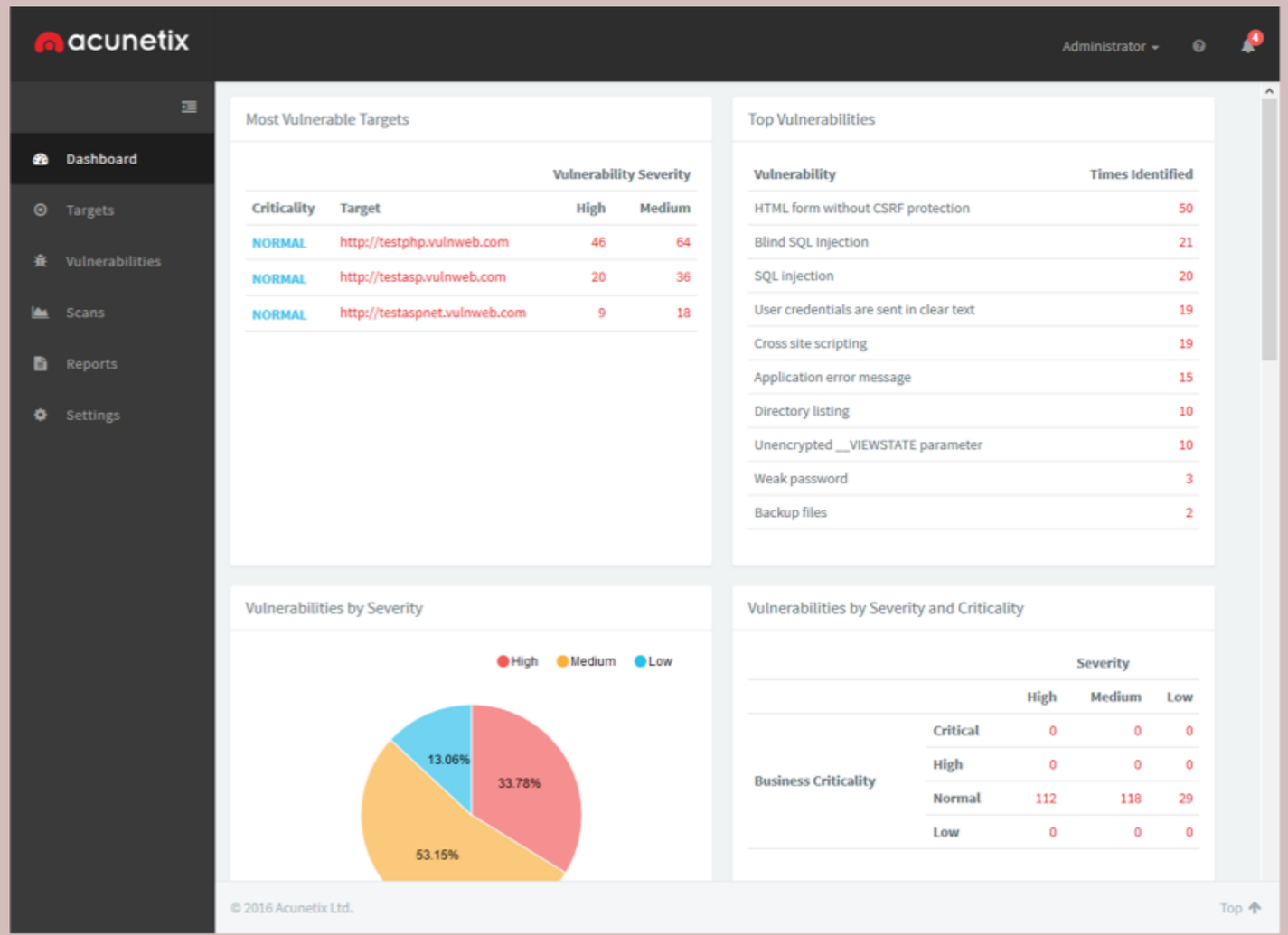
Security Architecture & Tool Sets

Web Application Vulnerability Scanning

- Dedicated web app vulnerability scanners do better than Nessus, Nexpose, and OpenVAS
- Identify problems with applications and the underlying web servers, databases, and infrastructure
- Examples
  - Acunetix WVS
  - Archni
  - Burp Suite
  - IBM's AppScan
  - HP's WebInspect
  - Netsparker
  - QualysGuard's Web Application Scanner
  - W3AF

JASON DION
TRAINING THE CYBER SECURITY WORKFORCE

Acunetix



JASON DION
TRAINING THE CYBER SECURITY WORKFORCE

Acunetix

SQL injection (verified)

Severity HIGH

Vulnerability description

This script is possibly vulnerable to SQL Injection attacks.

SQL injection is a vulnerability that allows an attacker to alter backend SQL statements by manipulating the user input. An SQL injection occurs when web applications accept user input that is directly placed into a SQL statement and doesn't properly filter out dangerous characters.

This is one of the most common application layer attacks currently being used on the Internet. Despite the fact that it is relatively easy to protect against, there is a large number of web applications vulnerable.

This vulnerability affects **/listproducts.php**.

Discovered by: Scripting (Sql_Injection.script).

Attack details

URL encoded GET input **artist** was set to **(select 1 and row(1,1)>(select count(*),concat(concat(CHAR(52),CHAR(67),CHAR(117),CHAR(98),CHAR(52),CHAR(117),CHAR(78),CHAR(77),CHAR(72),CHAR(79),CHAR(55)),floor(rand()*2))x from (select 1 union select 2)a group by x limit 1))**
Injected pattern found:

```
4Cub4uNMHO7
```

- View HTTP headers
- View HTML response
- Launch the attack with HTTP Editor
- Retest alert(s)
- Mark this alert as a false positive

The impact of this vulnerability

An attacker may execute arbitrary SQL statements on the vulnerable system. This may compromise the integrity of your database and/or expose sensitive information.

Depending on the back-end database in use, SQL injection vulnerabilities lead to varying levels of data/system access for the attacker. It may be possible to not only manipulate existing queries, but to UNION in arbitrary data, use subselects, or append additional queries. In some cases, it may be possible to read in or write out to files, or to execute shell commands on the underlying operating system.

Certain SQL Servers such as Microsoft SQL Server contain stored and extended procedures (database server functions). If an attacker can obtain access to these procedures it may be possible to compromise the entire machine.

How to fix this vulnerability

Your script should filter metacharacters from user input.
Check detailed information for more information about fixing this vulnerability.

Detailed information

- Click here for more detailed information about this vulnerability

Web references

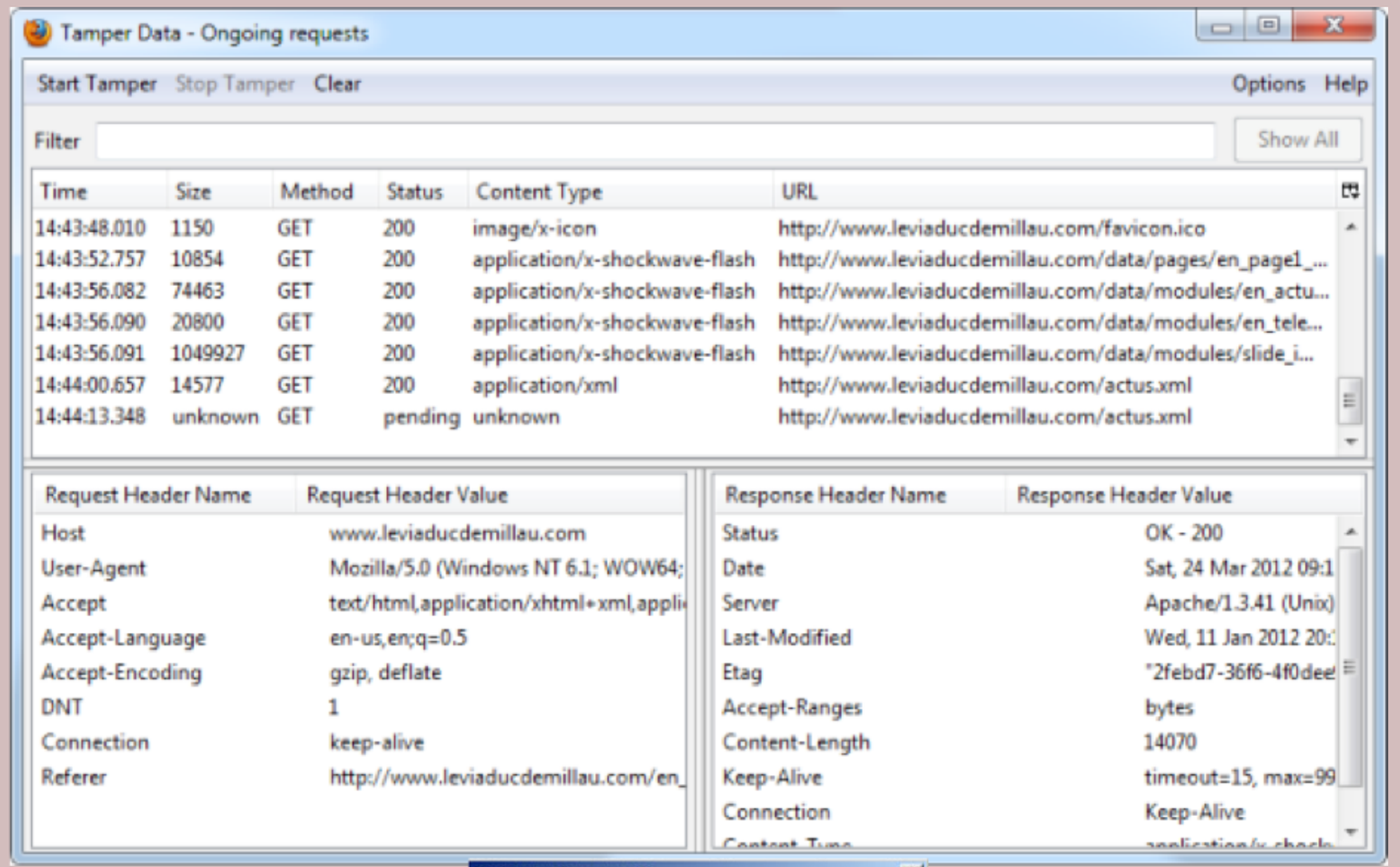
- Acunetix SQL Injection Attack
- Advanced SQL Injection
- Security Focus - Penetration Testing for Web Applications (Part Two)
- More Advanced SQL Injection

JASON DION
TRAINING THE CYBER SECURITY WORKFORCE

Manual Scanning

- Uses and interception proxy to capture communications between browser and server
- Testers can modify data sent and received
- Examples
  - Tamper Data for Firefox and Chrome
  - HttpFox
  - Fiddler
  - Burp Suite

Tamper Data

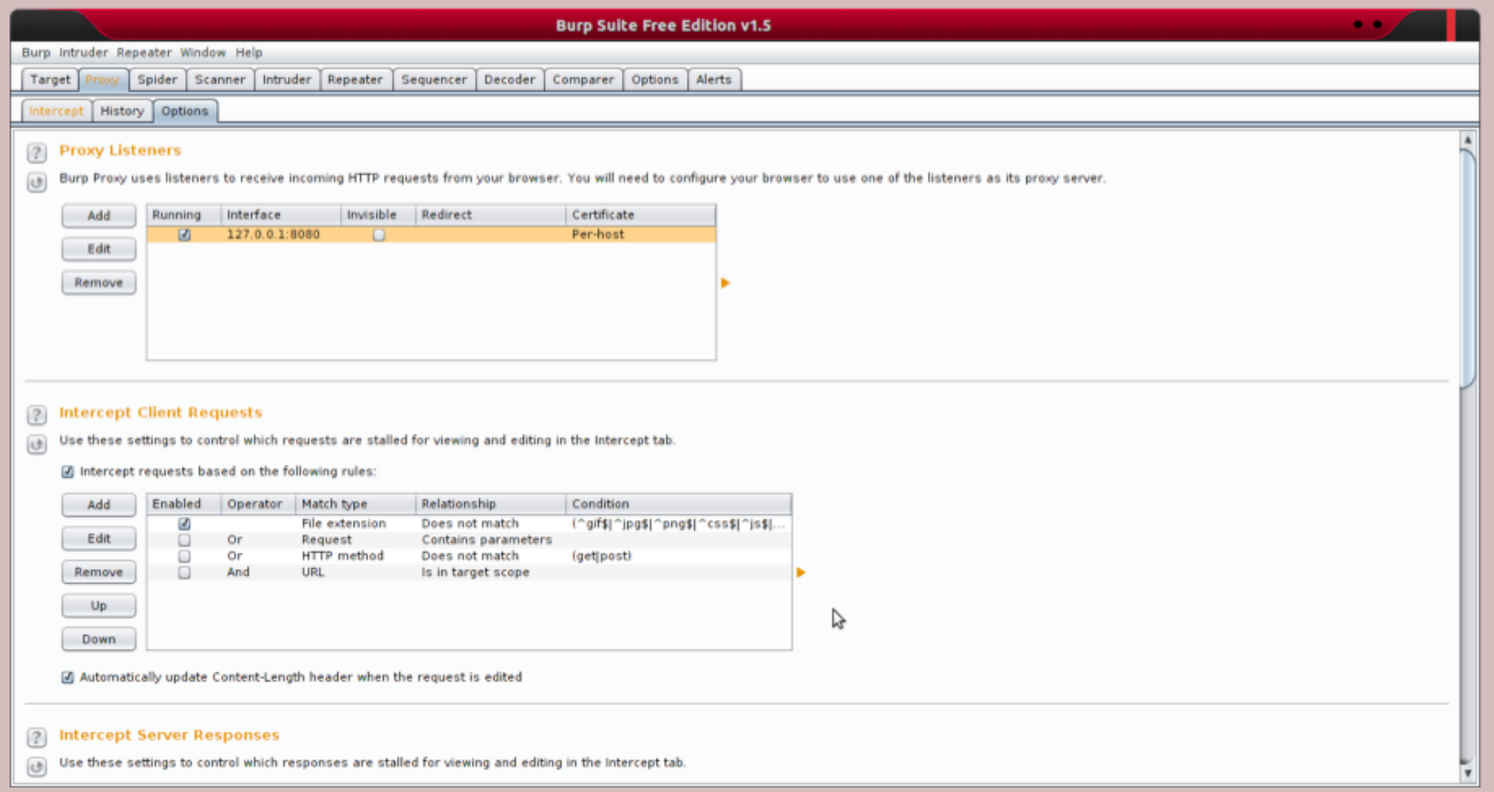


JASON DION

TRAINING THE CYBER SECURITY WORKFORCE

Burp Suite

- Automated and Manual modes



JASON DION
TRAINING THE CYBER SECURITY WORKFORCE

Outsource Your Scanning

- Even the best vulnerability scanners will miss business logic issues and other flaws
- Outsourcing to a security firm can identify issues that a web application scanner can't
- These firms can provide both static and dynamic analysis of your applications