Security Testing Course

W: www.sangbui.com

T: @sangsecurity

E: sang@sangbui.com

OWASP
Open Web Application
Security Project

Agenda

- Broken Authentication
- Broken Access Control
- Security Misconfiguration
- Using Components with Known Vulnerabilities
- Insufficient Logging & Monitoring
- Hands on Labs
- Q/A

OWASP Top 10

OWASP Top 10 - 2013	→	OWASP Top 10 - 2017	
A1 – Injection	→	A1:2017-Injection	✓
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication	
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure	✓
A4 – Insecure Direct Object References [Merged+A7]	∪	A4:2017-XML External Entities (XXE) [NEW]	
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]	
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration	
A7 – Missing Function Level Access Contr [Merged+A4]	∪	A7:2017-Cross-Site Scripting (XSS)	✓
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]	
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities	
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]	

OWASP Top 10

OWASP Top 10 - 2013	→	OWASP Top 10 - 2017	
A1 – Injection	→	A1:2017-Injection	✓
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication	✓
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure	✓
A4 – Insecure Direct Object References [Merged+A7]	∪	A4:2017-XML External Entities (XXE) [NEW]	
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]	✓
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration	✓
A7 – Missing Function Level Access Contr [Merged+A4]	∪	A7:2017-Cross-Site Scripting (XSS)	✓
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]	
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities	✓
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]	✓

Broken Authentication

It involves all kinds of flaws that are caused by error in implementations of authentication and session management

Broken Authentication

- Permits brute force or other automated attacks. No wrong password limit
- Permits default, weak, or well-known passwords
- Uses plain text, encrypted, or weakly hashed passwords
- Exposes Session IDs in the URL
- Does not rotate Session IDs after successful login
- Does not properly invalidate Session IDs.
- Indicate the username or password that was wrong when the login attempt fails
- Weak password change controls

Broken Authentication

Scenario #1: Airline reservations application supports URL rewriting, putting session IDs in the URL:

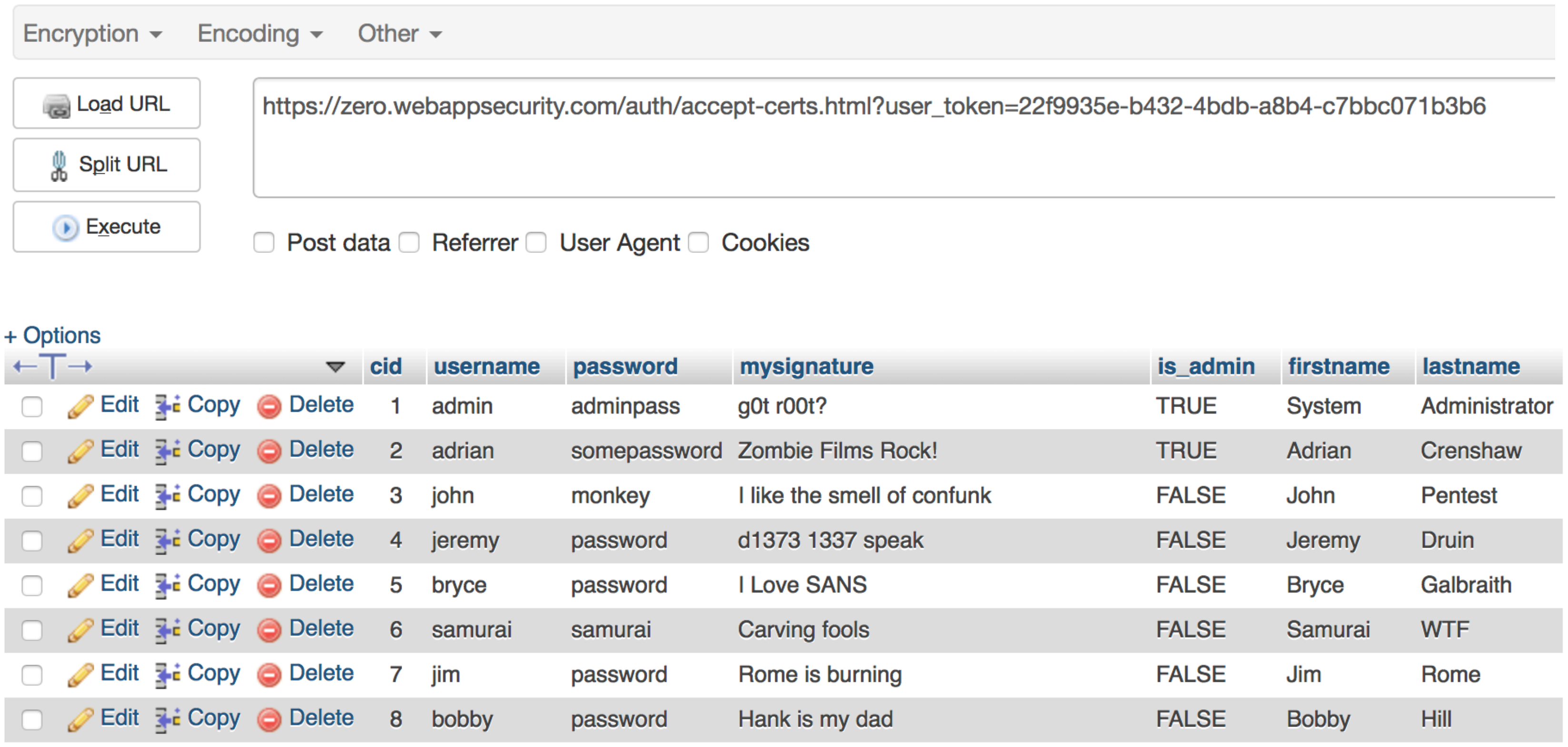
http://example.com/sale/saleitems?sessionid=268544541&dest=Hawaii

An authenticated user of the site wants to let his friends know about the sale. He e-mails the above link without knowing he is also giving away his session ID. When his friends use the link they will use his session and credit card.

Broken Authentication

Scenario #2: Application's timeouts aren't set properly. User uses a public computer to access site. Instead of selecting "logout" the user simply closes the browser tab and walks away. Attacker uses the same browser an hour later, and that browser is still authenticated.

Broken Authentication



```
https://zero.webappsecurity.com/auth/accept-certs.html?user_token=22f9935e-b432-4bdb-a8b4-c7bbc071b3b6
```

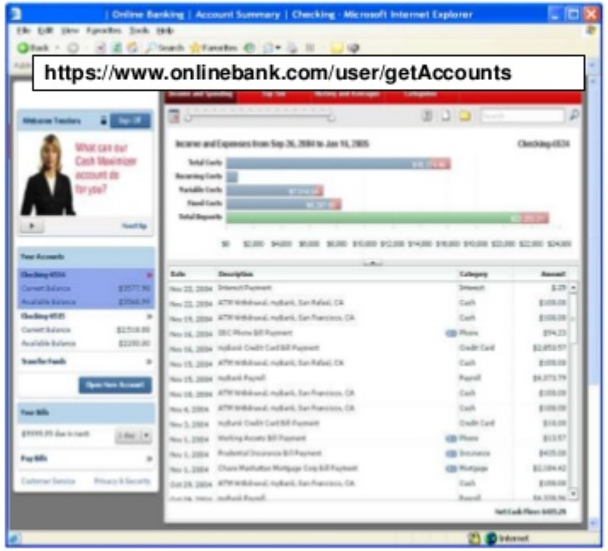
☐ Post data ☐ Referrer ☐ User Agent ☐ Cookies

+ Options

	cid	username	password	mysignature	is_admin	firstname	lastname
☐ Edit Copy Delete	1	admin	adminpass	g0t r00t?	TRUE	System	Administrator
☐ Edit Copy Delete	2	adrian	somepassword	Zombie Films Rock!	TRUE	Adrian	Crenshaw
☐ Edit Copy Delete	3	john	monkey	I like the smell of confunk	FALSE	John	Pentest
☐ Edit Copy Delete	4	jeremy	password	d1373 1337 speak	FALSE	Jeremy	Druin
☐ Edit Copy Delete	5	bryce	password	I Love SANS	FALSE	Bryce	Galbraith
☐ Edit Copy Delete	6	samurai	samurai	Carving fools	FALSE	Samurai	WTF
☐ Edit Copy Delete	7	jim	password	Rome is burning	FALSE	Jim	Rome
☐ Edit Copy Delete	8	bobby	password	Hank is my dad	FALSE	Bobby	Hill

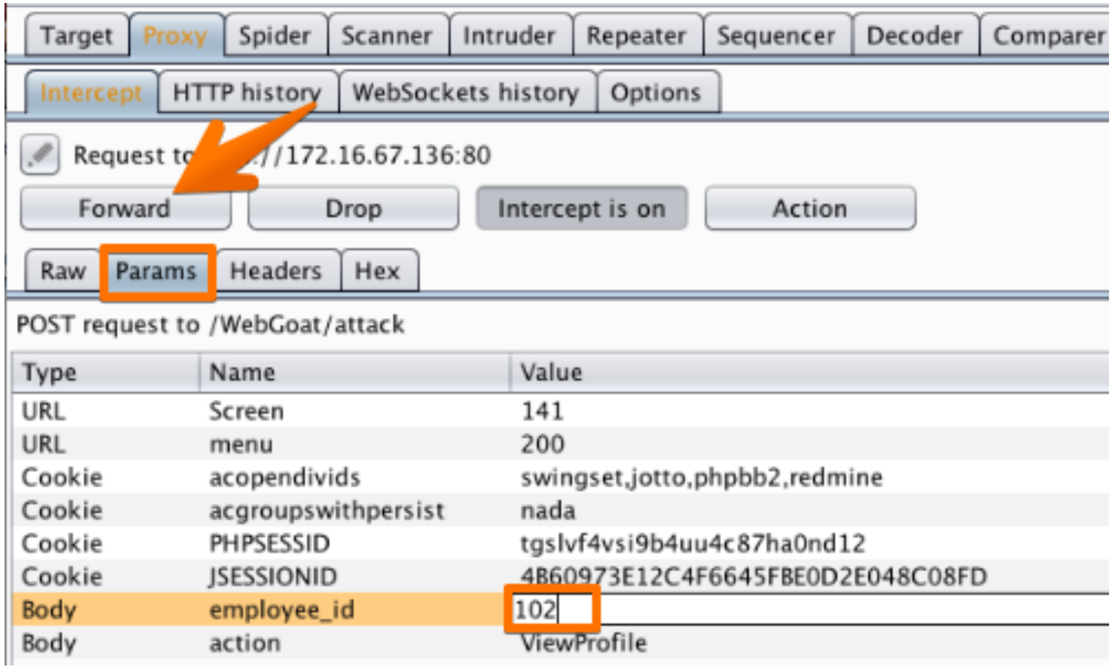
Broken Access Control

Missing Function Level Access Control Illustrated



- Attacker notices the URL indicates his role
  /user/getAccounts
- He modifies it to another directory (role)
  /admin/getAccounts, or
  /manager/getAccounts
- Attacker views more accounts than just their own

Broken Access Control



Request to ://172.16.67.136:80

POST request to /WebGoat/attack

Type	Name	Value
URL	Screen	141
URL	menu	200
Cookie	acopendivids	swingset,jotto,phpbb2,redmine
Cookie	acgroupswithpersist	nada
Cookie	PHPSESSID	tgslvf4vsi9b4uu4c87ha0nd12
Cookie	JSESSIONID	4B60973E12C4F6645FBE0D2E048C08FD
Body	employee_id	102
Body	action	ViewProfile

Another way is to identify user IDs and similar data in requests and simply change them to someone else.

Broken Access Control

Scenario #1: The attacker simply force browses to target URLs. The following URLs require authentication. Admin rights are also required for access to the admin_getappInfo page.

http://example.com/app/getappInfo
http://example.com/app/admin_getappInfo

If an unauthenticated user can access either page, that's a flaw. If an authenticated, non-admin, user is allowed to access the admin_getappInfo page, this is also a flaw.

Security Misconfiguration

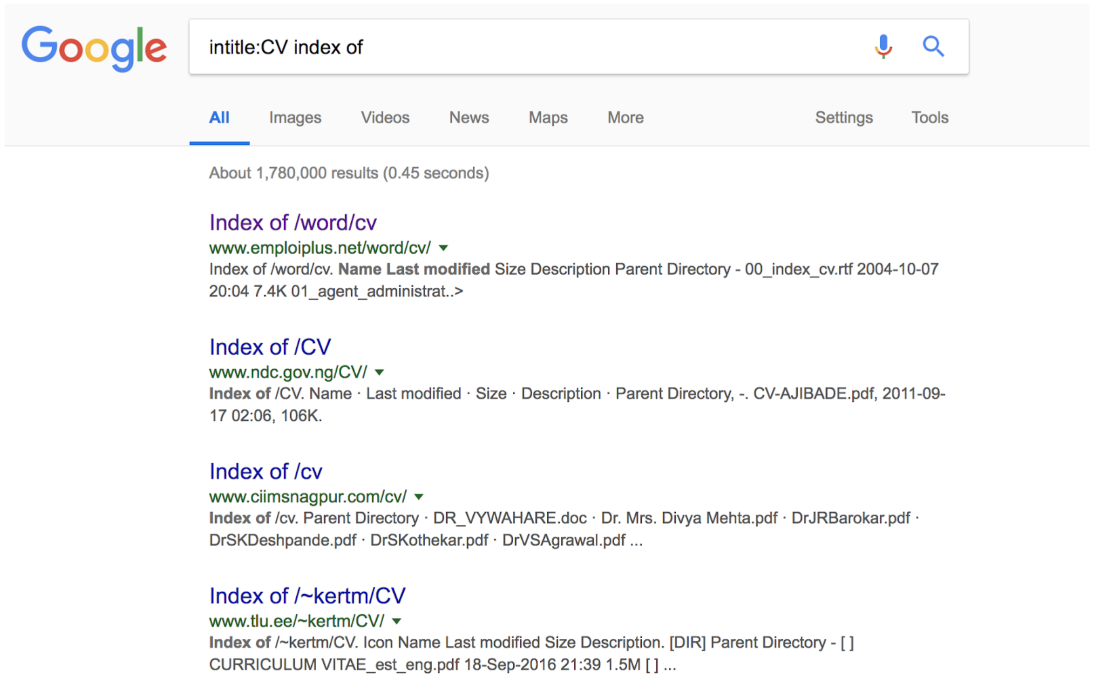
Weaknesses found in the configuration of web applications that may result in unintended application behaviour.

Security Misconfiguration

OWASP has identified 8 most probable security misconfiguration target areas that can be exploited by the hackers to compromise the security of web-based environments.

- Unpatched security flaws in the server software.
- Improper file and directory permissions.
- Unnecessary services in enabled state.
- Default accounts with their default passwords.
- Exposure of administrative or debugging notifications to general users.
- Misconfigured SSL certificates and encryption settings.
- Misconfiguration of user roles.
- Improper authentication with external systems.

Security Misconfiguration



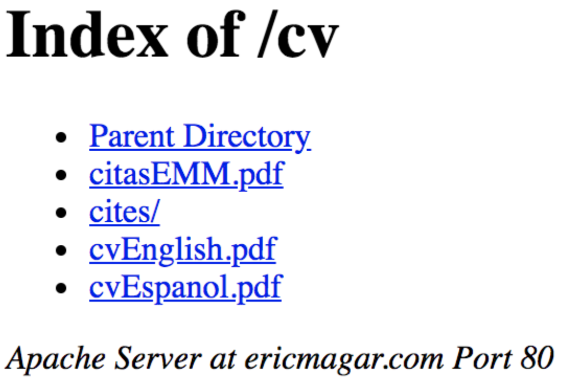
About 1,780,000 results (0.45 seconds)

Index of /word/cv
www.emploiplus.net/word/cv/
Index of /word/cv. **Name Last modified** Size Description Parent Directory - 00_index_cv.rtf 2004-10-07 20:04 7.4K 01_agent_administrat..>

Index of /CV
www.ndc.gov.ng/CV/
Index of /CV. Name · Last modified · Size · Description · Parent Directory, -. CV-AJIBADE.pdf, 2011-09-17 02:06, 106K.

Index of /cv
www.ciimsnagpur.com/cv/
Index of /cv. Parent Directory · DR_VYWAHARE.doc · Dr. Mrs. Divya Mehta.pdf · DrJRBarokar.pdf · DrSKDeshpande.pdf · DrSKothekar.pdf · DrVSAgrawal.pdf ...

Index of /~kertm/CV
www.tlu.ee/~kertm/CV/
Index of /~kertm/CV. Icon Name Last modified Size Description. [DIR] Parent Directory - [ ] CURRICULUM VITAE_est_eng.pdf 18-Sep-2016 21:39 1.5M [ ] ...

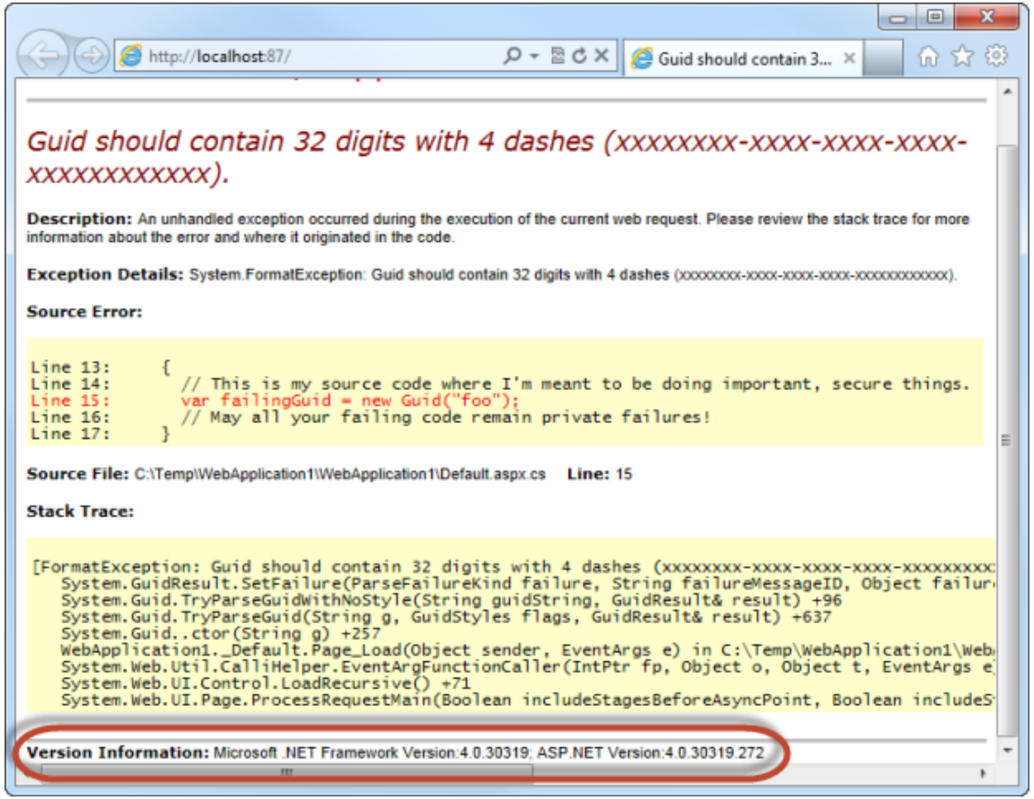


Index of /cv

- Parent Directory
- citasEMM.pdf
- cites/
- cvEnglish.pdf
- cvEspanol.pdf

Apache Server at ericmagar.com Port 80

Security Misconfiguration



Guid should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx).

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.FormatException: Guid should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx).

Source Error:

```
Line 13:     {
Line 14:       // This is my source code where I'm meant to be doing important, secure things.
Line 15:       var failingGuid = new Guid("foo");
Line 16:       // May all your failing code remain private failures!
Line 17:     }
```

Source File: C:\Temp\WebApplication1\WebApplication1\Default.aspx.cs   **Line:** 15

Stack Trace:

```
[FormatException: Guid should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxx
   System.GuidResult.SetFailure(ParseFailureKind failure, String failureMessageID, Object failur
   System.Guid.TryParseGuidWithNoStyle(String guidString, GuidResult& result) +96
   System.Guid.TryParseGuid(String g, GuidStyles flags, GuidResult& result) +637
   System.Guid..ctor(String g) +257
   WebApplication1._Default.Page_Load(Object sender, EventArgs e) in C:\Temp\WebApplication1\Web
   System.Web.Util.CalliHelper.EventArgFunctionCaller(IntPtr fp, Object o, Object t, EventArgs e
   System.Web.UI.Control.LoadRecursive() +71
   System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeS
```

Version Information: Microsoft .NET Framework Version:4.0.30319; ASP.NET Version:4.0.30319.272

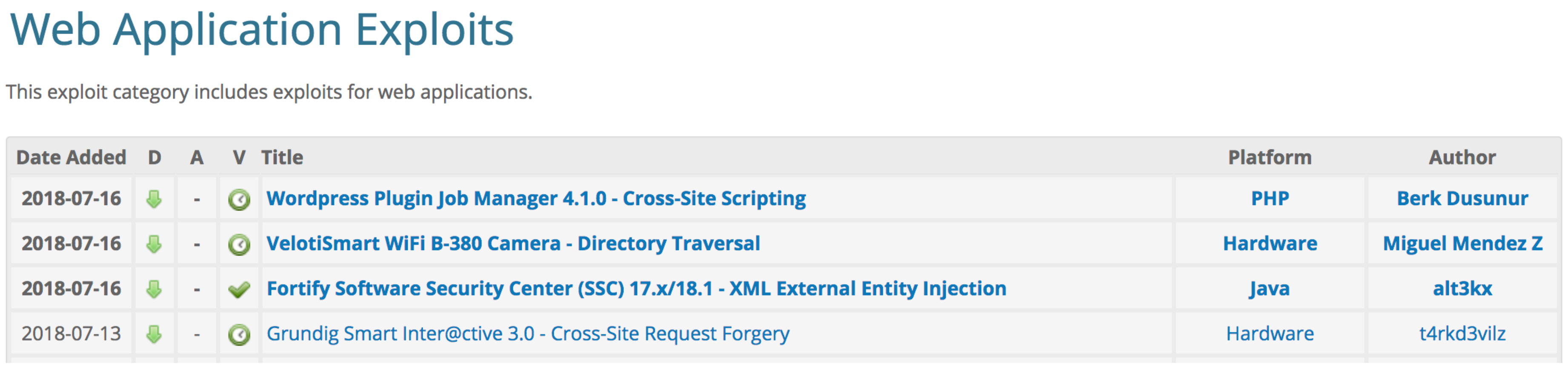
Using Components with Known Vulnerabilities

- It is very common for application to include a component with a known security vulnerability.
- The component with a known vulnerability could be the operating system itself, the CMS used, the web server, some plugin installed or even a library used by one of these plugins.

Using Components with Known Vulnerabilities

- Identify all the components or libraries the application uses and the versions.
- Monitor known security vulnerabilities in any published databases, project newsletters and mailing lists.
- Disablement of any functionality your application doesn’t require and any unnecessary aspects of the component

Using Components with Known Vulnerabilities



Web Application Exploits

This exploit category includes exploits for web applications.

Date Added	D	A	V	Title	Platform	Author
2018-07-16	⇩	-	◷	**Wordpress Plugin Job Manager 4.1.0 - Cross-Site Scripting**	**PHP**	**Berk Dusunur**
2018-07-16	⇩	-	◷	**VelotiSmart WiFi B-380 Camera - Directory Traversal**	**Hardware**	**Miguel Mendez Z**
2018-07-16	⇩	-	✓	**Fortify Software Security Center (SSC) 17.x/18.1 - XML External Entity Injection**	**Java**	**alt3kx**
2018-07-13	⇩	-	◷	Grundig Smart Inter@ctive 3.0 - Cross-Site Request Forgery	Hardware	t4rkd3vilz

Insufficient Logging & Monitoring

Basic vulnerabilities include:

- Unlogged events, such as failed login credentials
- Locally stored logs without cloud backup
- Misconfigurations in firewalls and routing systems
- Alerts and subsequent responses that aren’t handled effectively
- Malicious activity alerts not detected in real time

Q/A!

W: www.sangbui.com

T: @sangsecurity

E: sang@sangbui.com