



OWASP
AppSec EU
Belfast

8th to 12th
of May
2017

Waterfront
Conference
Center



DON'T GET CAUGHT EM-BED: FINDING AND PREVENTING VULNS AT ITS LOWEST LEVEL

Aaron Guzman

@SCRIPTINGXSS

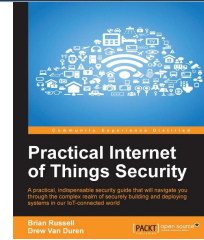


@SCRIPTINGXSS





#WHOAMI



Aaron Guzman

 @scriptingxss



DEFCON.  VILLAGE

HF

RSA® Conference



BSIDES



OWASP

cloud
CSA security
alliance

EMBEDDED DEVICES & TECHNOLOGY

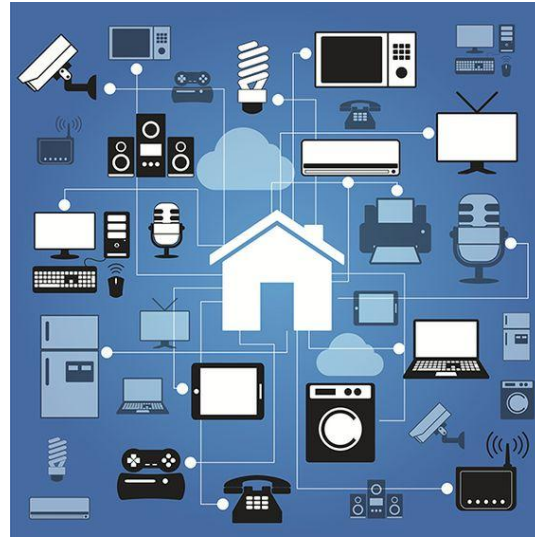
What does that even mean???





EMBEDDED == IOT

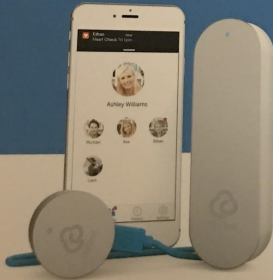
THE WORLD OF EMBEDDED





The smart way to manage your family's health.

Digital Thermometer +
Stethoscope
Thermomètre +
Stéthoscope Digital



Blocks Chronic Pain For Widespread Relief



TAKE READINGS PRENEZ DES RELEVÉS

CliniCloud enables you to record every cough, wheeze, fever or chill and save the data securely to your smartphone.

CliniCloud vous permet d'enregistrer chaque toux, rhume, fièvre ou grippe et sauvegarder ces données de manière sécurisée dans votre smartphone.

KEEP RECORDS SAUVEGARDEZ LES ENREGISTREMENTS

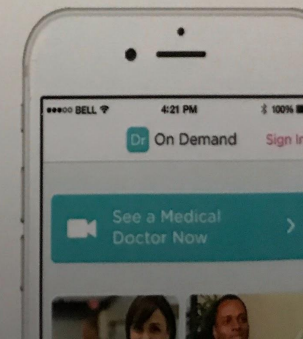
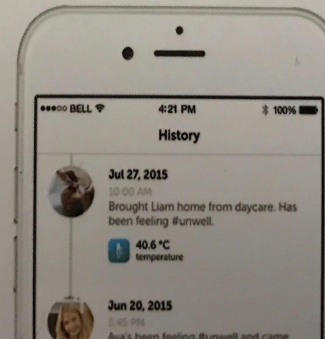
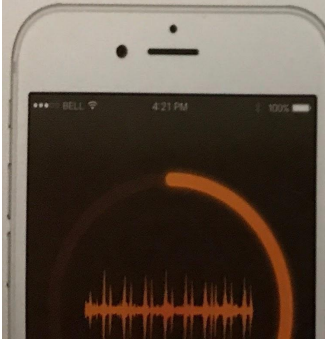
These recordings form personalized health records for you and your family and a secure way to remember everything.

Ces enregistrements vous permettent de créer un dossier de santé personnalisé pour vous et votre famille, d'une manière sécurisée pour ne jamais rien oublier.

GET ASSISTANCE OBTENEZ UN AVIS

If you ever have any concerns, you can send your recordings to a qualified physician and have them reviewed remotely.

En cas d'inquiétudes sur votre santé vous pouvez envoyer vos enregistrements à des médecins qualifiés pour qu'ils soient analysés par distance.



2016 EMBEDDED THREATS IN THE WILD

- ✖ Medical
 - Insulin pumps (CVE-2016-5084-6)
 - Clear Text Comms
- ✖ Consumer
 - More Backdoors
 - Command Injection
 - Mirai 🐛
 - Connected Vehicles
- ✖ Commercial
 - Cameras
 - Ransomware (CC Machines)
- ✖ Industrial Control Systems
 - Hardcoded Passwords



MIRAI WAS HUGE!!!!



Image provided by ThreatPost

400LB HACKER



HAJIME VS BRICKERBOT VS PERSIRAI



ATTACKERS BE LIKE



OWASP
AppSec EU
Belfast



ANOTHER ONE

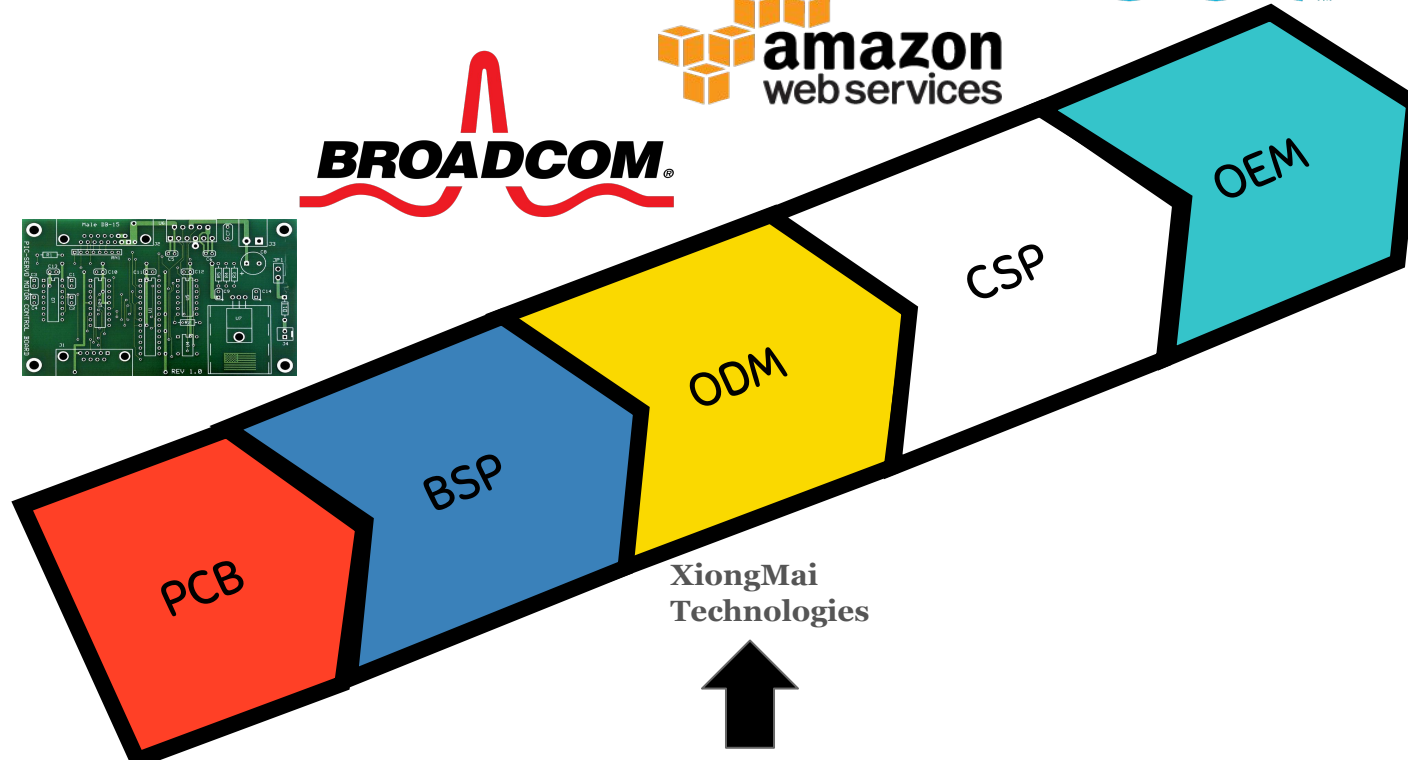
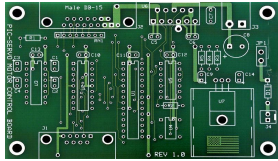
FOUND USING - FIRMWARE EXPLOITATION METHODOLOGY

1. Obtaining firmware
2. Analyzing firmware
3. Extracting the filesystem
4. Mounting file systems
5. Analyzing filesystem contents
6. Emulating firmware for dynamic & runtime analysis

<http://bit.ly/FirmwareAnalysisTools>

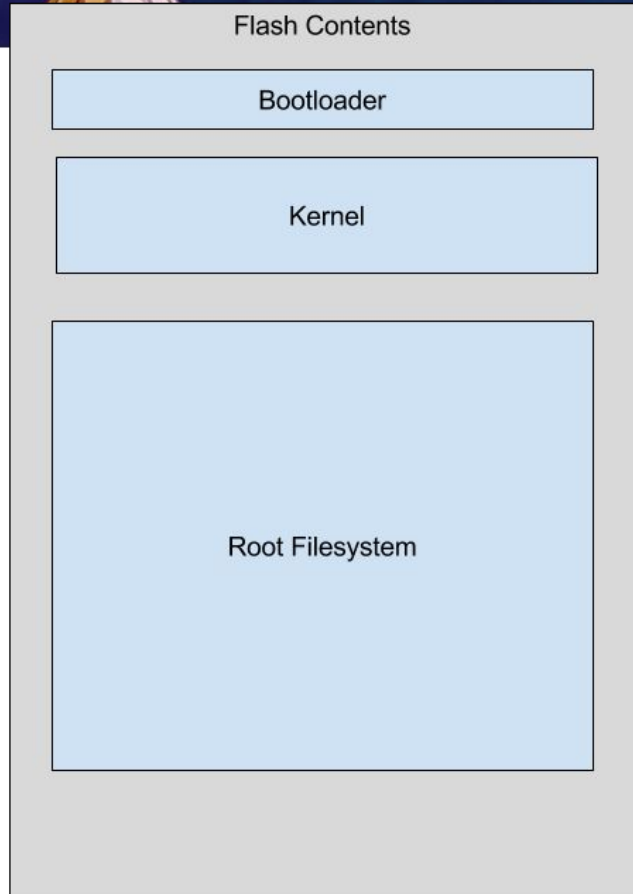
THE REASON WHY!

nest™



COMMON OPERATING SYSTEMS

- ✖ Embedded Linux
 - Android
 - Old..like, really old and New kernels
- ✖ Real Time Operating Systems (RTOS)
 - VxWorks
 - QNX (Blackberry)
 - MQX
 - Green Hills
- ✖ Windows Embedded 🖱
- ✖ Windows IoT Core



BEST PRACTICES TO SECURE EMBEDDED SOFTWARE

1. Buffer and Stack Overflow Protection
2. Injection Protections
3. Firmware Updates and Cryptographic Signatures
4. Securing Sensitive Information
5. Identity Management
6. Embedded Framework and C-Based Toolchain Hardening
7. Usage of Debugging Code and Interfaces
8. Transport Layer Security
9. Usage of Data Collection and Storage – Privacy
10. Third Party Code and Components

BEST PRACTICES TO SECURE EMBEDDED SOFTWARE

11. Threat Modeling



<https://scriptingxss.gitbooks.io/embedded-application-security-best-practices/>

INCUBATOR new projects

OWASP Embedded Application Security Project

Every year the prevalent use of embedded software within enterprise and consumer devices continues to rise exponentially. With widespread publicity of the Internet of Things (IoT), more and more devices are becoming network connected evidencing how essential it is to create secure coding guidelines for embedded software. Embedded Application Security is often not a high priority for embedded developers when they are producing devices such as routers, managed switches, medical devices, Industrial Control Systems (ICS), VoIP phones, IoT devices, and ATM Kiosks due to other challenges outside of development. Other challenges developers face may include, but are not limited to, the Original Design Manufacturer (ODM) supply chain, limited memory, a small stack, and the challenge of pushing firmware updates securely to an endpoint. The goals of this project are to create a list of best practices, provide practical guidance to embedded developers, and to draw on the existing OWASP resources that can bring application security expertise to the embedded world. It is important to note, each of the items and guidance points listed below are longstanding within software security. This document purely tailors issues that OWASP has previously provided guidance upon (e.g. OWASP Top 10, Mobile Top 10, etc.) to the embedded community. *Given the prevalence of Linux kernels utilized within embedded devices, all code examples are geared towards a POSIX environment but the principles are designed to be platform agnostic.*

Mailing List / Group Communication

[Embedded Sec Mailing List](#)
Please join our OWASP Slack channel; look for the [#embeddedappsec](#)

Project Leaders

[Aaron Guzman](#) @ @
[Alex Lafrenz](#) @ @

Related Projects

- [OWASP Internet of Things Project](#)
- [C-Based Toolchain Hardening](#)
- [OWASP Mobile Security Project](#)
- [IoT Firmware Analysis](#)

News and Events

Conferences that project leaders will be speaking at based upon the Embedded Application Security Project

- [07 April 2017] Sam Houston University (SHACS) Conference
- [07 April 2017] BSides Edinburgh
- [11 May 2017] AppSec EU
- [19 May 2017] HackMiami
- [23-26 May 2017] AusCERT
- [02 June 2017] SecurityFest

Releases

[Click here for version 1](#)
[PDF Version](#)

BUFFER AND STACK OVERFLOW PROTECTION

- ✘ Prevent usage of dangerous C Functions
 - `find . -type f -name '*.c' -print0|xargs -0 grep -e 'strcpy.strlen'|wc -l`
- ✘ Use safe equivalent functions for known vulnerable functions
 - `gets()` -> `fgets()`
- ✘ Ensure secure compiler flags or switches are utilized upon each firmware build. (e.g. `-fPIE`, `-fstack-protector-all`, `-Wl,-z,noexecstack`, `-Wl,-z,noexeccheap`)
- ✘ Enable stack protection in embedded build systems (Buildroot & Yocto)



Toolchain

Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> will exclude a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [] feature is excluded

```
Toolchain type (Buildroot toolchain) --->
*** Toolchain Buildroot Options ***
(buildroot) custom toolchain vendor name (NEW)
C library (uClibc-ng) --->
*** Kernel Header Options ***
Kernel Headers (Linux 4.8.x kernel headers) --->
*** uClibc Options ***
(package/uclibc/uClibc-ng.config) uClibc configuration file to use? (NEW)
() Additional uClibc configuration fragment files (NEW)
[ ] Enable RPC support (NEW)
[ ] Enable WCHAR support (NEW)
[ ] Enable toolchain locale/i18n support (NEW)
Thread library implementation (Native POSIX Threading (NPTL)) --->
[ ] Thread library debugging (NEW)
[*] Enable stack protection support
[*] Compile and install uClibc utilities (NEW)
[ ] Compile and install uClibc tests (NEW)
*** Binutils Options ***
Binutils Version (binutils 2.26.1) --->
() Additional binutils options
*** GCC Options ***
GCC compiler Version (gcc 5.x) --->
() Additional gcc options
[ ] Enable C++ support (NEW)
*** Fortran support needs a toolchain w/ wchar ***
[ ] Enable compiler link-time-optimization support (NEW)
[ ] Enable compiler OpenMP support (NEW)
[ ] Enable graphite support (NEW)
```

↑(+)


<Select>

< Exit >

< Help >

< Save >

< Load >



strncpy

WAS MADE FOR
memory

Why would you settle
for anything less?

strncpy can be hard to
resist, but the risks of
strncpy are unavoidable.
Only strncpy as a part of
security-aware development
is safe. Don't use strncpy!

You'll be glad you waited!



natashenka.ca/strncpy



Canadian Joke
Council

True Bugs Wait ♥

@natashenka
#truebugswait



Canadian Joke
Council

I respect myself. That's why I refuse to use `sprintf`.
Using `sprintf` is a decision you can never take back.
That's why I'm waiting until I'm older and there's a string
handling function that's right for me

Forget `sprintf`!



natashenka.ca/sprintf



True Bugs Wait ♥

@natashenka
#truebugswait

INJECTION PREVENTION

- ✘ Whitelist accepted commands
- ✘ Avoid utilizing user data into operation system commands
- ✘ Validate user input
- ✘ Context output encode characters
- ✘ [Commix](#)



/set_ftp.cgi?next_url=ftp.htm&loginuse
=admin&loginpas=&svr=192.16
8.1.1&port=21&user=ftp&pwd=\$(ping
%20192.168.1.3)&dir=/&mode=PORT&
u pload_interval=0

INJECTION EXAMPLE

```
enum { BUFFERSIZE = 512 };
```

Injection Payload:

```
any_cmd 'happy'; useradd  
'attacker'
```

```
void func(const char *input) {
```

```
    char cmdbuf[BUFFERSIZE];
```

```
    int len_wanted = snprintf(cmdbuf, BUFFERSIZE,  
                              "any_cmd '%s'", input);
```

```
    if (len_wanted >= BUFFERSIZE) {
```

```
        /* Handle error */
```

```
    } else if (len_wanted < 0) {
```

```
        /* Handle error */
```

```
    } else if (system(cmdbuf) == -1) {
```

```
        /* Handle error */
```

```
    }
```

```
}
```

FIRMWARE UPDATES

- ✘ Updates over TLS
- ✘ Automatic or scheduled updates
 - *Medical device cases
 - Force updates
- ✘ Anti-downgrade (anti-rollback) protections
- ✘ Cryptographically sign and verify updates
- ✘ Changelogs include security related vulnerabilities fixed
- ✘ Firmware versions are clearly displayed.

VERIFYING SIGNED PACKAGES

Downloading the kernel images

```
wget https://www.kernel.org/pub/linux/kernel/v4.x/linux-4.6.6.tar.xz  
wget https://www.kernel.org/pub/linux/kernel/v4.x/linux-4.6.6.tar.sign
```

Download the public key from a PGP keyserver in order to verify the signature.

```
# gpg2 --keyserver hkp://keys.gnupg.net --recv-keys 38DBBDC86092693E  
gpg: /root/.gnupg/trustdb.gpg: trustdb created  
gpg: key 38DBBDC86092693E: public key "Greg Kroah-Hartman (Linux kernel stable  
release signing key) <greg@kroah.com>" imported  
gpg: no ultimately trusted keys found  
gpg: Total number processed: 1  
gpg:             imported: 1
```

SECURING SENSITIVE INFORMATION

- ❌ Do not hardcode secrets such as:
 - passwords, usernames, tokens, private keys, PII or similar variants
- ❌ Use a security element (SE) or Trusted Execution Environment (TEE)
- ❌ Do not store secrets in an unprotected storage locations
 - EEPROM or other flash

**YOU KNOW WHAT REALLY
GRINDS MY GEARS?**



**COMMON SENSE ISN'T SO
COMMON ANYMORE**



OWASP
AppSec EU
Belfast

```
1  #define AUTH_OK 1
2  #define AUTH_FAIL -1
3
4  int alpha_auth_check(struct http_request_t *request)
5  {
6      if(strstr(request->url, "graphic/") ||
7          strstr(request->url, "public/") ||
8          strcmp(request->user_agent, "xmlset_roodkcableoj28840ybtide") == 0)
9      {
10         return AUTH_OK;
11     }
12     else
13     {
14         // These arguments are probably user/pass or session info
15         if(check_login(request->0xC, request->0xE0) != 0)
16         {
17             return AUTH_OK;
18         }
19     }
20
21     return AUTH_FAIL;
22 }
```



According to the [FTC's complaint](#), D-Link promoted the security of its routers on the company's website, which included materials headlined "EASY TO SECURE" and "ADVANCED NETWORK SECURITY." But despite the claims made by D-Link, the FTC alleged, the company failed to take steps to address well-known and easily preventable security flaws, such as:

- "hard-coded" login credentials integrated into D-Link camera software -- such as the username "guest" and the password "guest" -- that could allow unauthorized access to the cameras' live feed;



CVE-2017-8224 - BACKDOOR ACCOUNT

```
root:$1$ybdHbPDn$ii9aEIFNiolBbM  
9QxW9mr0:0:0::/root:/bin/sh
```

IDENTITY MANAGEMENT

- ✘ Over TLS*
- ✘ Password change upon installation**
- ✘ Separation of accounts for
 - internal web management
 - internal console access
 - remote web management
 - remote console access
- ✘ SessionIds and Cookies
 - Not in the URL
 - Secure and HttpOnly
 - Randomized and invalidated upon logout
- ✘ EEPROM & UART complex passwords

EMBEDDED FRAMEWORK HARDENING

- ✘ Services such as SSH have a secure password created.
- ✘ Remove unused language interpreters
- ✘ Remove dead code from unused libs
- ✘ Remove unused shell interpreters
- ✘ Remove legacy insecure daemons (Telnet, FTP, TFTP)
- ✘ Iterative threat models..please..thanks :)

Shell and utilities

Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> will exclude a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [] feature is excluded

```
*** Shells ***
[*] bash
[ ] dash
[ ] zsh
*** Utilities ***
[ ] at (NEW)
[ ] ccrypt (NEW)
[ ] dialog (NEW)
[ ] dtach (NEW)
[ ] file (NEW)
[ ] gnupg (NEW)
[ ] gnupg2 (NEW)
[ ] inotify-tools (NEW)
[ ] lockfile programs (NEW)
*** logrotate needs a toolchain w/ wchar ***
[ ] logsurfer (NEW)
[ ] pinentry (NEW) ----
*** ranger needs a toolchain w/ wchar, threads, dynamic library ***
[ ] screen (NEW)
[ ] sudo (NEW)
[ ] time (NEW)
[ ] tmux (NEW)
[ ] which (NEW)
[ ] xmlstarlet (NEW)
[ ] xxhash (NEW)
```

<Select>

< Exit >

< Help >

< Save >

< Load >



Networking applications

Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> will exclude a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [] feature is excluded

```
^( - )
[*] openssh
[ ] openswan (NEW)
[ ] openvpn (NEW)
[ ] p910nd (NEW)
[ ] phidgetwebservice (NEW)
*** portmap needs a toolchain w/ RPC ***
[ ] pound (NEW)
[ ] pppd (NEW)
[ ] pptp-linux (NEW)
[ ] privoxy (NEW)
[ ] proftpd (NEW)
[ ] proxychains-ng (NEW)
[ ] ptpd (NEW)
[ ] ptpd2 (NEW)
[ ] pure-ftpd (NEW)
[ ] putty (NEW)
[ ] quagga (NEW)
*** rabbitmq-server needs erlang ***
[ ] radvd (NEW)
[ ] rp-pppoe (NEW)
[ ] rpcbind (NEW)
[ ] rsh-redone (NEW)
[ ] rsync (NEW)
*** rtorrent needs a toolchain w/ C++, threads, wchar ***
[ ] rtptools (NEW)
*** samba4 needs a toolchain w/ RPC, wchar, dynamic library, threads ***
*** scons server needs a toolchain with dynamic library, C++, NPTL ***
[ ] ser2net (NEW)
```

<Select>

< Exit >

< Help >

< Save >

< Load >



USAGE OF DEBUGGING CODE AND INTERFACES

- ❌ Backdoor code with root privilege
 - Customer support
 - Debugging purposes
- ❌ Third party libraries, SDKs, and binary images need review
- ❌ Liability via service agreements to ODMs and third-parties
- ❌ Sooooooooooooooooooooo common 🙄

LOOKS GOOD

TO ME



OWASP
AppSec EU
Belfast

memegenerator.net

TRANSPORT LAYER SECURITY

- ✘ Use TLS 1.2 (or highest possible)
- ✘ Validate the certificate public key, hostname, and chain.
- ✘ Ensure new certificates and their chains use SHA256 for signing
- ✘ Disable deprecated SSL and early TLS versions.
- ✘ Disable deprecated, NULL and weak cipher suites.
- ✘ Ensure proper certificate update features are available upon expiration.
- ✘ Verify TLS configs with `nmap --script ssl-enum-ciphers.nse`, `TestSSLServer.jar`, `ssllscan` and/or `sslyze`

**You're using SSL. But have you
checked the certificate?**



Using SSL is always safer than not using it. But to protect against most attacks, it's important to validate the certificate. Your certificate could be expired, or provided by a malicious attacker!

No SSL is Never Safe

Test:~ nmap --script ssl-enum-ciphers.nse 192.168.1.1 -p 443



OWASP
AppSec EU
Belfast

```
Starting Nmap 7.12 ( https://nmap.org ) at 2017-02-28 20:28 PST
Nmap scan report for [REDACTED] (192.168.1.1)
Host is up (0.0018s latency).
PORT      STATE SERVICE
443/tcp   open  https
| ssl-enum-ciphers:
|   SSLv3:
|     ciphers:
|       TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (dh 1024) - D
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_DES_CBC_SHA (dh 1024) - D
|       TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 1024) - D
|       TLS_RSA_WITH_AES_128_CBC_SHA (rsa 1024) - A
|       TLS_RSA_WITH_AES_256_CBC_SHA (rsa 1024) - A
|       TLS_RSA_WITH_DES_CBC_SHA (rsa 1024) - D
|       TLS_RSA_WITH_IDEA_CBC_SHA (rsa 1024) - A
|       TLS_RSA_WITH_RC4_128_MD5 (rsa 1024) - A
|       TLS_RSA_WITH_RC4_128_SHA (rsa 1024) - A
|     compressors:
|       DEFLATE
|       NULL
|     cipher preference: client
|     warnings:
|       CBC-mode cipher in SSLv3 (CVE-2014-3566)
|       Ciphersuite uses MD5 for message integrity
|       Weak certificate signature: SHA1
|   TLSv1.0:
|     ciphers:
|       TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (dh 1024) - D
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_DES_CBC_SHA (dh 1024) - D
|       TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 1024) - D
|       TLS_RSA_WITH_AES_128_CBC_SHA (rsa 1024) - A
|       TLS_RSA_WITH_AES_256_CBC_SHA (rsa 1024) - A
|       TLS_RSA_WITH_DES_CBC_SHA (rsa 1024) - D
|       TLS_RSA_WITH_IDEA_CBC_SHA (rsa 1024) - A
|       TLS_RSA_WITH_RC4_128_MD5 (rsa 1024) - A
|       TLS_RSA_WITH_RC4_128_SHA (rsa 1024) - A
|     compressors:
|       DEFLATE
|       NULL
|     cipher preference: client
|     warnings:
|       Ciphersuite uses MD5 for message integrity
|       Weak certificate signature: SHA1
|   _ least strength: D
Nmap done: 1 IP address (1 host up) scanned in 0.71 seconds
```

USAGE OF DATA COLLECTION AND STORAGE

- ✘ Privacy-by-design
 - Acquire only data for business and/or operational purpose.
- ✘ Transparency by including details on information being collected, stored, and distributed via privacy policies.
- ✘ Allow the device owner to reset or remove their personal data before transfer to another user or destruction.

Reset advertising ID

Opt out of interest-based ads

Instruct apps not to use your advertising ID to build profiles or show you interest-based ads.



Ads by Google

Your advertising ID:
cd0d6448-eae9-4ba7-a24a-3719ebdb5693



ALL ADVERTISERS

Limit Ad Tracking



[Reset Advertising Identifier...](#)



Opt out of receiving ads targeted to your interests. You may still receive the same number of ads, but the ads may be less relevant to you.

ADVERTISING IN APPLE APPS

[View Ad Information](#)

View the information used by Apple to deliver more relevant ads to you in Apple News and the App Store. Your personal data is not provided to third-parties.

THIRD PARTY CODE AND COMPONENTS

- ✘ Bill of materials
- ✘ Check against vulnerabilities DBs
- ✘ Loads of free tools to help
 - Retirejs - JavaScript
 - LibScanner - Yocto Build
 - NSP - NodeJS 
 - Lynis - OS hardening..
 - OWASP ZAP - Web Testing 
- ✘ Review changelogs of toolchains, software packages, and libraries
- ✘ Utilize package managers (opkg, ipkg, rpm etc...) or custom update mechanisms for misc libraries

```
# ./cli.py --format yocto  
  "path/to/installed-packages.txt"  
dbs/ > cve_test.xml
```

```
#tail cve_test.xml
```

```
<failure> Medium (6.8) - Use-after-free vulnerability in libxml2 through 2.9.4,  
as used in Google Chrome before 52.0.2743.82, allows remote attackers to cause  
a denial of service or possibly have unspecified other impact via vectors  
related to the XPointer range-to function.
```

```
CVE Published on: 2016-07-23
```

```
https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-5131 </failure>
```

```
</testcase>
```

```
<testcase id="CVE-2016-9318" name="CVE-2016-9318" classname="libxml2 - 2.9.4"  
time="0">
```

```
<failure> Medium (6.8) - libxml2 2.9.4 and earlier, as used in XMLSec 1.2.23 and  
earlier and other products, does not offer a flag directly indicating that the  
current document may be read but other files may not be opened, which makes it  
easier for remote attackers to conduct XML External Entity (XXE) attacks via a  
crafted document.
```

```
CVE Published on: 2016-11-15
```

```
https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-9318 </failure>
```

```
</testcase>
```

```
</testsuite>
```

RINSE & REPEAT

- ✘ Continuous threat models
- ✘ Continuous testing
- ✘ Update update update
- ✘ Disclosure policy (ISO 29147)
- ✘ Involvement in the community



OWASP
AppSec EU
Belfast

Closing Thoughts



IT'S STILL A ODAY



OWASP
AppSec EU
Belfast

IF THEY DIDN'T UPDATE

imgflip.com

LIVE



OWASP
AppSec EU
Belfast


DMCA security research exemption for consumer devices

By: Aaron Alva | Oct 28, 2016 2:12PM

TAGS: [Data security](#) | [Office of Technology Research and Investigation \(OTRI\)](#) | [Research](#)



With the stroke of a pen, the Librarian of Congress has authorized security researchers who are acting in good faith to conduct controlled research on consumer devices so long as the research does not violate other laws such as the Computer Fraud and Abuse Act (CFAA). This temporary exemption to the Digital Millennium Copyright Act (DMCA) begins today. The new temporary exemption is a big win for security researchers and for consumers who will benefit from increased security testing of the products they use.

The [Digital Millennium Copyright Act \(DMCA\)](#)  makes it illegal to circumvent controls that prevent access to copyrighted material. The result is that under the DMCA, researchers can't investigate and discover security vulnerabilities if doing so requires reverse engineering or circumventing controls such as obfuscated code. The Librarian of Congress can adopt exemptions to the DMCA's anti-circumvention statute for various technologies. These exemptions have allowed individuals to unlock tablets and wearables, jailbreak mobile devices, circumvent brand-specific 3D ink restrictions on 3D printers, and more. Exemptions take away a legal hurdle and help protect conduct without fear of legal recourse. It is important to note that the rule requires a careful setup and testing environment in order to fall under the good faith security research exemption, and does not exempt researchers from other laws such as the CFAA.

This blog post describes some of the basics of the DMCA security research exemption, and possible avenues of security research that relate to consumer devices.

What type of research environment does the exemption require?

There are at least four main requirements researchers must meet when setting up a research environment in order to fall under the exemption. First, the computer program, or any devices on which those programs run, must be "lawfully acquired." Second, during research, the device and computer program should operate "solely for the purpose of good-faith security research." This means, in part, that the research "must be conducted in a controlled setting designed to avoid harm to individuals or the public." Third, the research must not begin before today, [October 28, 2016](#).



THANK YOU!!!

QUESTIONS? □

AARON GUZMAN

AARON.GUZMAN AT OWASP.ORG