Mitigación de riesgos en software de terceros en Google

Google

Staff Information Security Lead
fjserna@[gmail.com | google.com]
@fjserna

Responsable de ISE-TPS (prod security) y sus diferentes sub-grupos.
- Third party code security
- Vulnerability research
- Exploit mitigations
- Sandboxing technologies
- API hardening
- ...

Código “third party” y su seguridad en la empresa

It's an open-source world: 78 percent of companies run open-source software

- More than 55 percent of respondents said their company has no formal policy or procedure for open-source use. Moreover, only 27 percent have a formal policy for employee contributions to OSS projects.
- More than 50 percent are not satisfied with their ability to understand known security vulnerabilities in open-source components, and only 17 percent plan to monitor open source code for security vulnerabilities.

http://www.zdnet.com/article/its-an-open-source-world-78-percent-of-companies-run-open-source-software/

Common Vulnerabilities and Exposures

The Standard for Information Security Vulnerability Names

GitHub Security Alerts

Vulnerability coverage

Vulnerabilities that have CVE IDs (publicly disclosed vulnerabilities from the National Vulnerability Database) will be included in security alerts. However, not all vulnerabilities have CVE IDs—even many publicly disclosed vulnerabilities don't have them. We'll continue to get

Pregunta #1: ¿Cuántas vulnerabilidades han sido documentadas (CVE) en **openssl** durante 2017?

https://www.openssl.org/news/vulnerabilities.html
http://www.cvedetails.com/vulnerability-list/vendor_id-217/Openssl.html

Pregunta #2: ¿Cuántas vulnerabilidades han sido documentadas (CVE) en **ffmpeg** durante 2017?

https://ffmpeg.org/security.html
https://www.cvedetails.com/vulnerability-list/vendor_id-3611/Ffmpeg.html

Ejemplos de vulnerabilidades en código “third party”

CVE-2015-7547 GLIBC dns getaddrinfo vulnerability, stack overflow

```
[[root@sandbox-3]$ gcc -o client client.c
[[root@sandbox-3]$
[[root@sandbox-3]$ ./client
 Segmentation fault (core dumped)
[[root@sandbox-3]$
[[root@sandbox-3]$ wget https://google.com
 --2016-02-16 15:51:51--  https://google.com/
 Resolving google.com... Segmentation fault (core dumped)
[[root@sandbox-3]$
[[root@sandbox-3]$ curl https://google.com
 Segmentation fault (core dumped)
[[root@sandbox-3]$
 [root@sandbox-3]$
```

CVE-2017-14491 - dnsmasq DNS heap overflow

```
@@ -1113,36 +1130,47 @@ int add_resource_record(struct dns_header *header, char *limit, int *truncp, int
 #endif

       case '4':
+        CHECK_LIMIT(INADDRSZ);
         sval = va_arg(ap, char *);
         memcpy(p, sval, INADDRSZ);
         p += INADDRSZ;
         break;

       case 'b':
+        CHECK_LIMIT(1);
         usval = va_arg(ap, int);
         *p++ = usval;
         break;

       case 's':
+        CHECK_LIMIT(2);
         usval = va_arg(ap, int);
         PUTSHORT(usval, p);
         break;

       case 'l':
+        CHECK_LIMIT(4);
         lval = va_arg(ap, long);
         PUTLONG(lval, p);
         break;
```

CVE-2017-14491 - dnsmasq DHCP stack overflow

```
@@ -206,6 +206,9 @@ static int dhcp6_maybe_relay(struct state *state, void *inbuff, size_t sz,
    /* RFC-6939 */
    if ((opt = opt6_find(opts, end, OPTION6_CLIENT_MAC, 3)))
      {
+       if (opt6_len(opt) - 2 > DHCP_CHADDR_MAX) {
+         return 0;
+       }
        state->mac_type = opt6_uint(opt, 0, 2);
        state->mac_len = opt6_len(opt) - 2;
        memcpy(&state->mac[0], opt6_ptr(opt, 2), state->mac_len);
```

Source: Ken White - https://twitter.com/kennwhite/status/699623000191045632/photo/1?ref_src=twsrc%5Etfw

Recap

- Uso común de código “third_party” en la empresa
- Problema de inventario
- No todas las vulnerabilidades están etiquetadas
- “License does not dictate code quality”

Preguntas que los CISOs se deben hacer:

¿Tenemos un proceso de respuesta ante vulnerabilidades?
¿Cómo mitigamos esas vulnerabilidades públicas no etiquetadas?
¿Y las aún por descubrir/publicar?
¿Siguen los proyectos “third party” rigurosos procesos de seguridad?

¿Que hacemos en Google?

Contribuciones “open source” de Google en el campo de la seguridad

Google Authenticator

Open source version of Google Authenticator (except the Android app)

BoringSSL

A fork of OpenSSL used to implement cryptography and TLS across most of Google's products

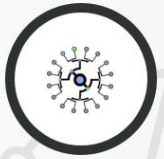
Vendor Security Assessment Questionnaire...

An interactive questionnaire to assess 3rd party security programs

syzkaller

coverage-guided Linux system call fuzzer

Certificate Transparency



A framework for monitoring and auditing SSL certificates in nearly real time

YARA

YARA : The pattern matching swiss knife

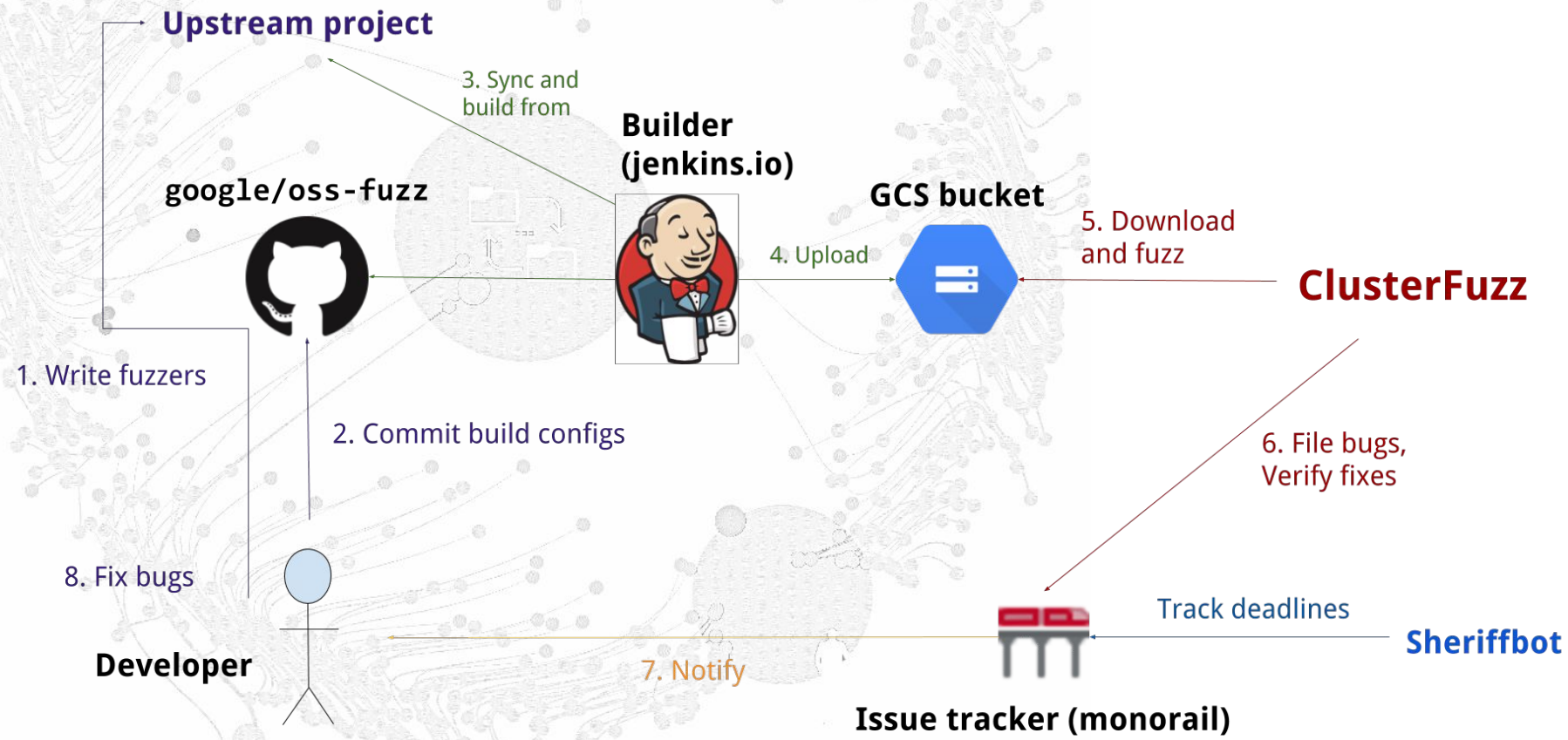
Estrategia para mitigar los riesgos del uso de código “third party”

Policy & Processes

- Inventario
  - Política de que es (o no) aceptable
  - Proceso de uso de nuevo código “third_party”
    - Seguridad por defecto: fuzzing, sandboxing, ...
- Monitorización CVE
  - Escalabilidad → automatismos
  - SLO
- Concepto de co-responsabilidad

Vulnerability Research

Además de los iniciativas internas: glibc, dnsmasq,... OSS Fuzz



Mitigations

Scudo user mode allocator (LLVM -fsanitize=scudo)

Objetivo: hacer más difícil la explotación de “heap based vulnerabilities”

```
void main(void)
{
  char *p;
  p = new char[100]; std::cout << (void *)p << std::endl;
  delete[] p;
  p = new char[100]; std::cout << (void *)p << std::endl;
  delete[] p;
  delete[] p;
  p = new char[100]; std::cout << (void *)p << std::endl;
  p = new char[100]; std::cout << (void *)p << std::endl;
  return 0;
}
```

system malloc:

```
0x7f1349ceb010
0x7f1349ceb010
*** Error in ...: double free or
corruption (fasttop):
0x00007f1349ceb010 ***
Aborted (core dumped)
```

Scudo allocator:

```
0x7b82f5928110
0x7b82f5926490
ERROR: invalid chunk state when
deallocating address
0x7b82f5926490
```

Mitigations

Linux Kernel research and security development

Objetivo: hacer más difícil la escalada de privilegios o "sandbox escape"

index : kernel/git/torvalds/linux.git

Linux kernel source tree

about summary refs log tree commit diff stats

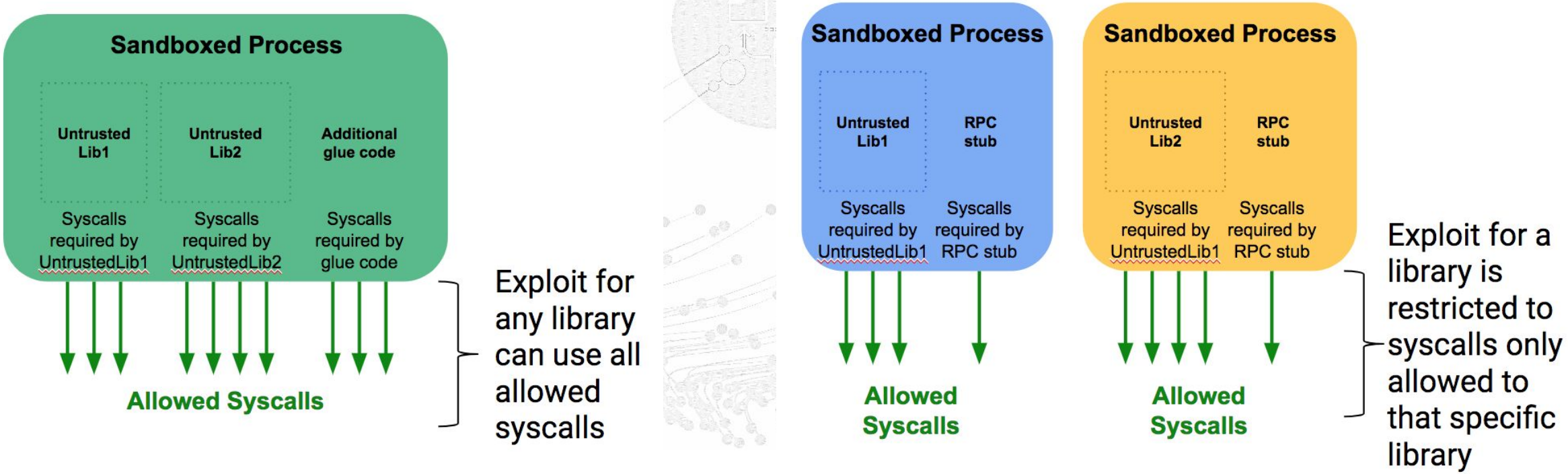
Age	Commit message (Expand)
2017-09-17	arm64/syscalls: Move address limit check in loop
2017-09-17	arm/syscalls: Optimize address limit check
2017-09-17	Revert "arm/syscalls: Check address limit on user-mode return"
2017-09-17	syscalls: Use CHECK_DATA_CORRUPTION for addr_limit_user_check
2017-07-08	arm64/syscalls: Check address limit on user-mode return
2017-07-08	arm/syscalls: Check address limit on user-mode return
2017-07-08	x86/syscalls: Check address limit on user-mode return
2017-03-21	x86/headers: Simplify asm/fixmap.h inclusion into asm/pgtable*.h
2017-03-18	x86/mm: Correct fixmap header usage on adaptable MODULES_END
2017-03-16	x86: Make the GDT remapping read-only on 64-bit
2017-03-16	x86: Remap GDT tables in the fixmap section

Containment

Objetivo: Limitar acciones en procesos comprometidos

Publicación en 2018:

- seccomp-bpf based sandbox framework
- sandboxed api framework



- **E-Mails**
  - info@ccn-cert.cni.es
  - ccn@cni.es
  - sat-inet@ccn-cert.cni.es
  - sat-sara@ccn-cert.cni.es
  - organismo.certificacion@cni.es
- **Websites**
  - www.ccn.cni.es
  - www.ccn-cert.cni.es
  - www.oc.ccn.cni.es
- **Síguenos en**