

KEYPATCH: binary patcher for IDA Pro

<http://keystone-engine.org/keypatch>

NGUYEN Anh Quynh <aquynh -at- gmail.com>

Trada hacking - 16/9/2016

Who am I

- **Nguyen Anh Quynh**, aquynh-at-gmail.com
 - ▶ Nanyang Technological University, Singapore
 - ▶ PhD in Computer Science
 - ▶ Operating System, Virtual Machine, Binary analysis, etc
 - ▶ Capstone disassembler: <http://capstone-engine.org>
 - ▶ Unicorn emulator: <http://unicorn-engine.org>
 - ▶ Keystone assembler: <http://keystone-engine.org>



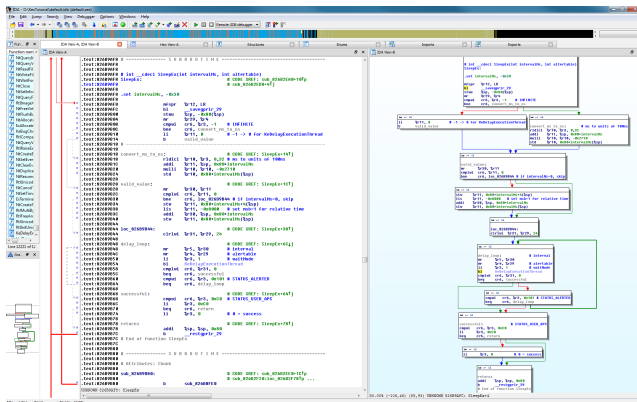
Binary patching

- CrackMe, CTF challenges
- Malware analysis
- Modify binary without source code :-)

URLZone Banking Trojan

```
00414ef2 ffd1      call    ecx
00414ef4 83f800    cmp     eax,0
00414ef7 7408      je      image00400000+0x14f01 (00414f01)
00414ef9 6a00      push    0
00414efb ffd3      call    ebx {ntdll!RtlExitUserThread (77a5f608)}
00414efd a1fcbacc59 mov     eax,dword ptr ds:[59CCBAFCh]
00414f02 5b       pop     ebx
```

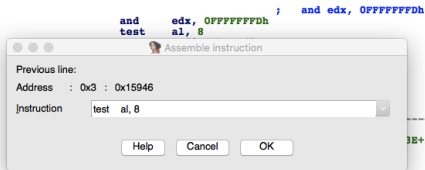
- <https://www.hex-rays.com>
- De-facto binary analysis tool
- Extendable with plugin SDK (C, Python)



Built-in binary patcher of IDA

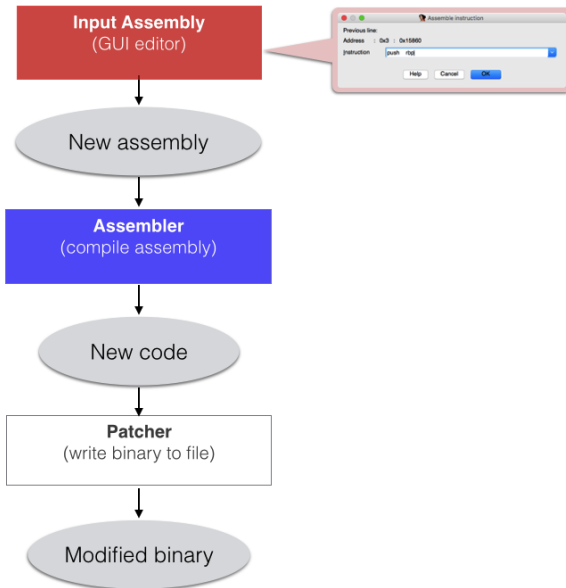
- Modify binary code with menu "Edit | Patch program | Assemble..."

```
0015941
0015943 83 E2 FD
0015946 A8 08
0015948 88 57 10
001594B 74 2C
001594D 8B 47 08
0015950 3B 47 14
0015953 7D 05
0015955 89 47 14
0015958 EB 08
001595A
001595A
001595A
001595A 3B 47 1C
001595D 7E 03
001595F 89 47 1C
0015962
```

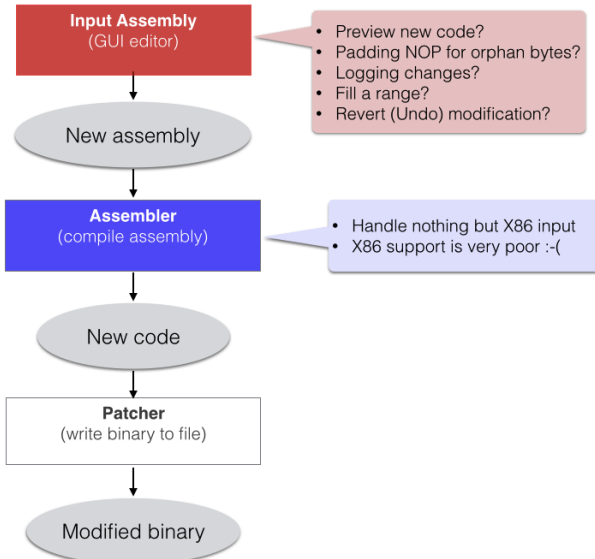


- Save changes permanently to binary file
 - ▶ Menu "Edit | Patch program | Apply patches to input file..."

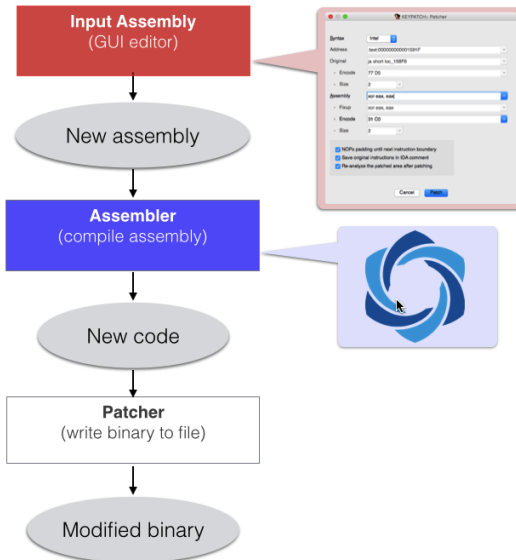
How it work?



Issues of IDA built-in binary patcher



Keypatch Solution



Keystone == Next Generation Assembler Framework



Assembler framework

Definition

- Compile assembly instructions & returns encoding as sequence of bytes
 - ▶ Ex: `inc EAX` → 40
- May support high-level concepts such as macro, function, etc
- Framework to build apps on top of it

Applications

- Dynamic machine code generation
 - ▶ Binary rewrite
 - ▶ Binary searching

Good assembler framework?

- True framework
 - ▶ Embedded into tool without resorting to external process
- Multi-arch
 - ▶ X86, Arm, Arm64, Mips, PowerPC, Sparc, etc
- Updated
 - ▶ Keep up with latest CPU extensions
- Multi-platform
 - ▶ *nix, Windows, Android, iOS, etc
- Bindings
 - ▶ Python, Ruby, Go, NodeJS, etc

Existing assembler frameworks

- Nothing is up to our standard, even in 2016!
 - ▶ Yasm: X86 only, no longer updated
 - ▶ Intel XED: X86 only, miss many instructions & closed-source
 - ▶ Other important archs: Arm, Arm64, Mips, PPC, Sparc, etc?

Life without assembler frameworks?

- People are very much struggling for years!
 - ▶ Use existing assembler tool to compile assembly from file
 - ▶ Call linker to link generated object file
 - ▶ Use executable parser (ELF) to parse resulted file for final encoding
- Ugly and inefficient
- Little control on the internal process & output
- Cross-platform support is very poor

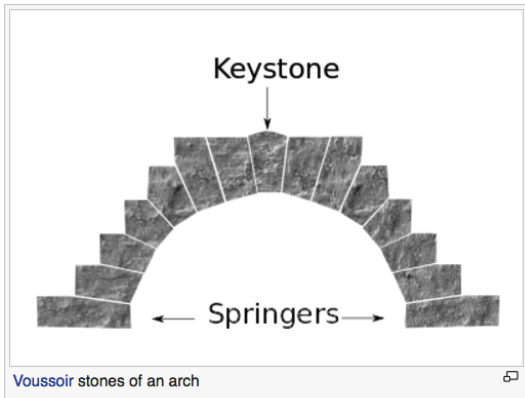
"If not now, then when? If not you, then who?" - Kailash Satyarthi

Keystone (architecture)

From Wikipedia, the free encyclopedia

This article is about the architectural element. For other uses, see [Keystone \(disambiguation\)](#).

A **keystone** is the wedge-shaped stone piece at the apex of a masonry arch, the generally round one at the apex of a [vault](#). In both cases it is the final piece placed during construction and locks all the stones into position, allowing the arch or vault to bear weight.^{[1][2][3]} In both arches and vaults, keystones are often enlarged beyond the structural requirements, and often decorated in some way. Keystones are often placed in the centre of the flat top of openings such as doors and windows, essentially for decorative effect.



Timeline

- Indiegogo campaign started on March 17th, 2016 (for 3 weeks)
 - ▶ 99 contributors, 4 project sponsors
- Beta code released to beta testers on April, 2016
 - ▶ Only Python binding available at this time
- Version 0.9 released on May, 2016: <http://keystone-engine.org>
 - ▶ More bindings by beta testers: NodeJS, Ruby, Go & Rust
- Version 0.9.1 released on July 27th, 2016
 - ▶ 2 more bindings: Haskell & OCaml

Keystone engine

- True framework
 - ▶ Embedded into tool without resorting to external process
- Multi-arch
 - ▶ X86, Arm, Arm64, Mips, PowerPC, Sparc, Hexagon, SystemZ
- Updated
 - ▶ Keep up with latest CPU extensions
- Multi-platform
 - ▶ *nix, Windows, Android, iOS, etc
- C++ core & multi-bindings
 - ▶ Python, Ruby, Go, NodeJS, OCaml, Rust, Haskell
- Support various X86 undocumented instructions
- Compact & lightweight: 10x smaller than LLVM

Keypatch binary patcher for IDA

Keypatch

- Co-developed with Thanh Nguyen (VNSecurity.net)
- Open source IDA plugin <http://keystone-engine.org/keypatch>
- Tool for assembling & patching in IDA
- Built on top of Keystone assembler framework
 - ▶ Version 1.0 released at BlackHat USA 2016, August 4th, 2016
 - ▶ Version 2.0 released on September 14th, 2016
 - ▶ Version 2.0.1 released on September 15th, 2016

Keypatch - Patcher

KEYPATCH:: Patcher

Syntax: Intel

Address: .text:000000000001591F

Original: ja short loc_158F6

- Encode: 77 D5

- Size: 2

Assembly: xor eax, eax

- Fixup: xor eax, eax

- Encode: 31 C0

- Size: 2

☒ NOPs padding until next instruction boundary

☒ Save original instructions in IDA comment

☒ Re-analyze the patched area after patching

Cancel Patch

Keypatch - Fill Range

KEYPATCH:: Fill Range

Syntax: Intel

Start: .text:0000000000015910

End: .text:0000000000015915

Size: 6

Assembly: nop

- Fixup: nop
- Encode: 90
- Size: 1

☒ NOPs padding until next instruction boundary

☒ Save original instructions in IDA comment

☒ Re-analyze the patched area after patching

Cancel Patch

Keypatch - Assembler

KEYPATCH:: Assembler

Arch: ARM

Endian: Big Endian

Address: .text:00000000000015913

Assembly: nop

- Fixup: nop

- Encode: E3 20 F0 00

- Size: 4

Cancel

Keypatch vs IDA's built-in patcher

- More friendly
 - ▶ Code preview
 - ▶ Padding NOPs automatically
 - ▶ Logging modifications
 - ▶ Fill a range of selected code
 - ▶ Assembler (do not modify)
 - ▶ Revert (undo)
- Support 8 architectures
 - ▶ Arm, Arm64, Hexagon, Mips, PowerPC, Sparc, SystemZ, X86
 - ▶ X86 support is fantastic
- Open source

Conclusions

- **Keypatch** is a superior binary patcher for IDA
 - ▶ Multi-arch + multi-platform
 - ▶ Feature-rich & friendly
 - ▶ Open source
- Looking for new contributors for our open source projects
 - ▶ Keypatch + Keystone engine
 - ▶ Capstone engine + Unicorn engine



References

- Keypatch: <http://keystone-engine.org/keypatch>
- Keystone assembler
 - ▶ Homepage: <http://keystone-engine.org>
 - ▶ Twitter: [@keystone_engine](#)
 - ▶ Github: <http://github.com/keystone-engine/keystone>
 - ▶ Mailing list: <http://freelists.org/list/keystone-engine>

Questions and answers

KEYPATCH: binary patcher for IDA Pro

<http://keystone-engine.org/keypatch>

NGUYEN Anh Quynh <aquynh -at- gmail.com>