TOOLSMITHING AN IDA BRIDGE: A TOOL BUILDING CASE STUDY

Adam Pridgen

Matt Wollenweber

Presentation Agenda

- Motivation and Purpose
- Toolsmithing
- Identifying the short-cuts to meet project needs
- Processes for Expediting Development
- Prototyping, Modifying, Testing, Restart?!?
- Extension development with WinDbg
- Idabridge demonstration

Introductions: Adam

- TODO Add pertinent Information
- Who I am.
- What I have done.
- Where I am going.

Introductions: Matt

- TODO Add pertinent Information
- Who I am.
- What I have done.
- Where I am going.

Motivation and Purpose

- Learn and teach methods for developing tools
- Introduce toolsmithing to those interested in tool development
- Discuss what we learned from implementing our tool
- Release an Alpha version of our idabridge

Toolsmithing

- Toolsmithing is the process of making tools
- Tools can be in any space
- Generally, not a standalone application
- Ranges from short scripts to full blown libraries
- Focus on utility not usability
- Takes on the following forms
  - X is needed to make Y create widgets
  - Z needs to be built, but nothing exists currently

Toolsmithing Tools

- High Level Languages (Python or Ruby)
- HL Programming Environments (iPython)
- Debuggers (PDB, WinDbg, Olly, etc.)
- Network Sniffers for network debugging
- Books and code lying around the home or net
- Anything that gets the job done fast

Our Toolsmithing Process

- Building is Believing
- Loner Development Squads
- The World is Big Chances are it exists
- Don’t reinvent the wheel, steal one
- KISS your tools they love you

Building is Believing

- Good tools are not built overnight
  - Sometimes maybe
- Build it once to get an idea
- Build it again because the 2nd time shine
- Third time is a charm
- More than one implementation is likely
  - idabridge's cmd handling took 3 iterations
- Build to what is needed now

Loner Development Squads

- Creating Milestones
  - Milestones should aggregate into something
  - Keep milestones small when developing alone
  - Keep a friend (esp one who cares) on speed dial
- Writing concise and re-usable
  - Think about what is being developed
  - Make it abstract and re-usable
  - Time is critical, if you can think of anything, just go

The World is Big…

- Open Source is the best source for help
- Code can be reviewed and repurposed
- Existing code is fantastic for real-world examples
- Documentation and APIs don’t run in debuggers
- Implementing complex components
  - Building a fuzzer, take someone elses protocol impl.
  - Building a DNS Mapping tool, use BIND for the DNS

Introducing idabridge

- Extensible network listener for IDA Pro
- Gives IDA users a “remote control”
- Implements a async. Network listener
- Provides extensibility using a Python Class
- Aims to be a middleware layer for other tools:
  - Binary Diffing
  - Debuggers
  - Other frameworks such as Radare

Current State of Tings

- Users are moving to “cloud” based solutions
- Collaboration among analysts and users
- Federation of data
  - Moving data from whatever to wherever
- Heterogenous tool chests and chains
- Employers and contracts
  - Cool tools are developed, but may not leave closed environments

Goals and Challenges

- Investigate cloud based reversing tools
- Evaluate the feasibility for a middleware for our current tools
- Determine what tools will make a difference
- Future direction for supporting technologies
  - Cloud based Python Interpreter
  - Migration of Binaries and environment for analysis

Idabridge Components

- IDA Pro networking client
- WinDbg network server
- Python environment Exported from IDAPython
- Command Handler for Debuggers and IDA Pro
  - VDB/Vtrace
  - WinDbg
  - IDA Pro

Tools Used for Development

- Visual Studio for C/C++ on Windows
  - Debugging a debugger?!?
  - IDE
- iPython & Python
  - Used to create scripts to write code and classes
  - Functional code testing
  - Data manipulation and verification
  - Server mock-ups to test the initial cmd handling

Development Environments

- Windows 7, 64-bit
  - VS 2010
  - IPython

- Windows XP VM, 32-bit
  - VS 2010
  - …

Overall Lessons Learned

- Debugging Debuggers
- Write Scripts to Implement code
  - Parsing IDAPython APIs
  - Implementing Python Command Handlers
  - Writing Long Logic C++ Statements
  - Creating Stub Functions

Toolsmithing: Research Phase

- Initial Research and Development: 90 Hours
  - Researching code and capabilities (IDA Pro and WinDbg)
  - Learning APIs and how to use them
  - Planning, Testing, Adjusting
  - Includes Coding and Testing
- Created a GUI to simulate a debugger
- Implemented IDA Commands Manually Using C++ only
- Implemented Separate Command Handling on Platforms
- Mostly “Get it working phase”

Toolsmithing: Research Phase

- Lessons Learned
  - Write scripts to write code and functions
  - Wrote a “dumb” server to send and reply to msg.s
  - Documentation is not your friend find examples
  - Find examples that have been repeated

Toolsmithing: Phase 2

- Defcon Talk accepted, resumed development
- Development: 60 Hours (2 weeks)
  - Developed an Abstract Cmd Handler Based on Names
  - Included Typed Argument Marshaling (str, int, long, byte)
  - Combined the Network Stack and Handling
- Never tested and threw out most of the code
- Realized atm there was no added value
- Breakthrough was the command handling
- Combined source and functionality

Toolsmithing: Cmd Handler

- Development: 30 Hours (1.5 weeks)
  - Developed the abstract handler
  - Added IDAPython Bridge to the mix
- Figured out how to add IDA Python Bridging

Toolsmithing: Cmd Handler

- Development: 20 Hours
  - Added Python as the Main Command Handling
  - Co-Developed Vtrace/VDB command handling

Idabridge Demonstration

Conclusions

- Creativity, Patience, Persistence, and Tenacity
- Motivation relies on small milestones
- Expectations are limited by time frame
- Tool Code quality != production CQ
- <FINAL PROJECT Data>

Idabridge information

- Special Thanks To:
  - Praetorian
  - C. Eagle and T. Vidas for Collabreate
  - E. Erdelyi for IDAPython
  - Pusscat / Lin0xx for Byakugan

- Code URL
  - http://TBD
- Presentation URL
  - http://TBD

Questions & Comments

- Adam.pridgen@[thecoverofnight.com || praetorian.com]
- mjw@cyberwart.com