Using the Metasploit Hardware Bridge to Attack Non-Ethernet Systems

Craig Smith
Research Director of Transportation Security at Rapid7
Agenda

• Overview of what the HW Bridge is
• Details on how it works
• How you can build hardware to support Metasploit
• How you can write modules for supported hardware
• Newest patches
• Future tech
Metasploit Hardware Bridge

- Most Popular FOSS Penetration Tool
- Full Integration
- Hardware Independent
- Scriptable
- Works against any type of Hardware
- Current Extensions: CAN Bus, RF Transceivers, Zigbee
What does it work on?

• FOSS Hardware with Networking
• FOSS Hardware w/o networking (Serial, USB, Other?)
• Proprietary Tools that want to integrate with MSF
• Proprietary Tools that have never heard of MSF
Project Goals

• Needs to work as a standalone

• Needs to work with a red team

• Needs to be useful for internal security teams and Q&A
Non-Ethernet Examples

MSF Relay
“Metasploit Compatible” Devices

MSF Relay
HW Bridge API

http://OpenGarages.org/hwbridge/

```json
{
  "hw_speciality": {
    "automotive": true,
  },
  "hw_capabilities": {
    "can": true,
    "j1939": true
  }
}
```
"Methods": [
{
  "method_name": "display_message",
  "method_desc": "Displays a message on the LCD, scrolls if message is too large",
  "args": [
    {
      "arg_name": "msg",
      "arg_type": "String",
      "required": true
    }
  ]
}
Local HTTP Relay Server

MSF Relay

Security Team
Hardware Devs

- No need to know Metasploit
- API (Relay) can be written in any language
- Support whatever you can from opengarages.org/hwbridge
Making of a Relay

# User modules for connecting
• modules/auxiliary/server/local_hwbridge.rb # Example server relay
• modules/auxiliary/client/hwbridge/connect.rb

# External Relays, ELM327, Killerbee
• tools/hardware

# Other places to find relays
• The core source repo. Example: rfcat
Changing the UI Commands

# Hardware Bridge UI Extension
• lib/rex/post/hwbridge
• lib/rex/post/hwbridge/extensions/automotive
• lib/rex/post/hwbridge/ui/console/command_dispatcher/automotive.rb

# Hardware Bridge API for scripting modules
• lib/msf/core/post/hardware/automotive/
Scripting API

- lib/msf/core/post/hardware/

```python
can1 18DB33F1 [8] 02 01 00 00 00 00 00 00
can1 18DAF118 [8] 06 41 00 98 18 00 01 AA
can1 18DAF110 [8] 06 41 00 BE 3E A8 13 00

pids = get_current_data_pids(canbus, src, dst, options)
```
Porting RFCat

# RFCat

d.setMdmModulation(MOD_ASK_OOK)
d.setFreq(results.baseFreq)
d.setMdmSyncMode(0)
d.setMdmDRate(results.baudRate)
d.setMdmChanSpc(24000)
d.setModeIDLE()
d.setPower(results.power)

# MSF

set_modulation("ASK/OOK")
set_freq(datastore['FREQ'])
set_sync_mode(0)
set_baud(datastore['BAUD'])
set_channel_spc(24000)
set_mode("idle")
set_power(datastore['POWER'])
Most Recent Additions

- ISO-TP Flow Control Support
- ISO-TP Detailed Padding Support
- NordicRF Transceiver Chipset Support
Future Development

LOTS of stuff!
Below is the short list of near-term stuff:
• Additional CAN Protocol SDKs, TP 2.0, Better J1939, etc.
• Other Bus protocols, K-Line, VPW, LIN
• Full SDR Support (Soapy)
• Lots of new modules (Airbags, Keyless entry)
• Additional HW support (LAWICEL 2.0)

Where can you help?
• Share your tests/modules with the community
• Help build a standard test suite
• Include a relay with your project
Demo & Questions