Dissecting Smart Meters

Justin Searle
Managing Partner – UtiliSec

Architectural Overview of Smart Meters

AMI Network Diagram

AMI DMZ Network

Alternate Names:
- AMI Perimeter Network

Backhaul

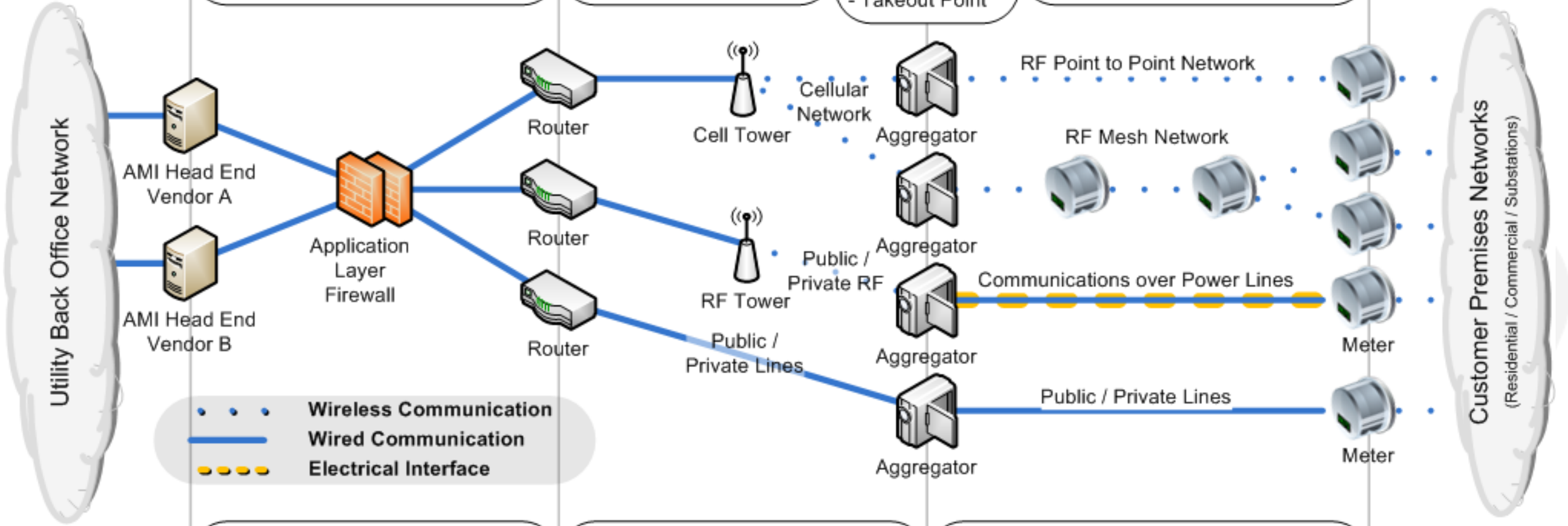
Alternate Names:
- Wide Area Network (WAN)

Aggregator

Alternate Names:
- Collector
- Base Station
- Access Point
- Takeout Point

Neighborhood Area Network (NAN)

Alternate Names:
- Field Area Network



AMI DMZ Protocols

Application Layer (OSI 5-7):
- ANSI C12.22
- ANSI C12.18 / C12.19 / C12.21

Transport Layer (OSI 4):
- TCP / UDP

Network Layer (OSI 3):
- IP4 / IP6

PHY / MAC Layter (OSI 1-2):
- Ethernet

Backhaul Protocols

Application Layer (OSI 5-7):
- ANSI C12.22
- ANSI C12.18 / C12.19 / C12.21

Transport Layer (OSI 4):
- TCP / UDP

Network Layer (OSI 3):
- IP4 / IP6

PHY / MAC Layter (OSI 1-2):
- Radio (WiMax 802.16d/e)
- Cellular (EVDO / GPRS / …)
- Power Line Carrier (BPL / P1901)
- Fiber (RFoG / FTTP / Ethernet)

NAN Protocols

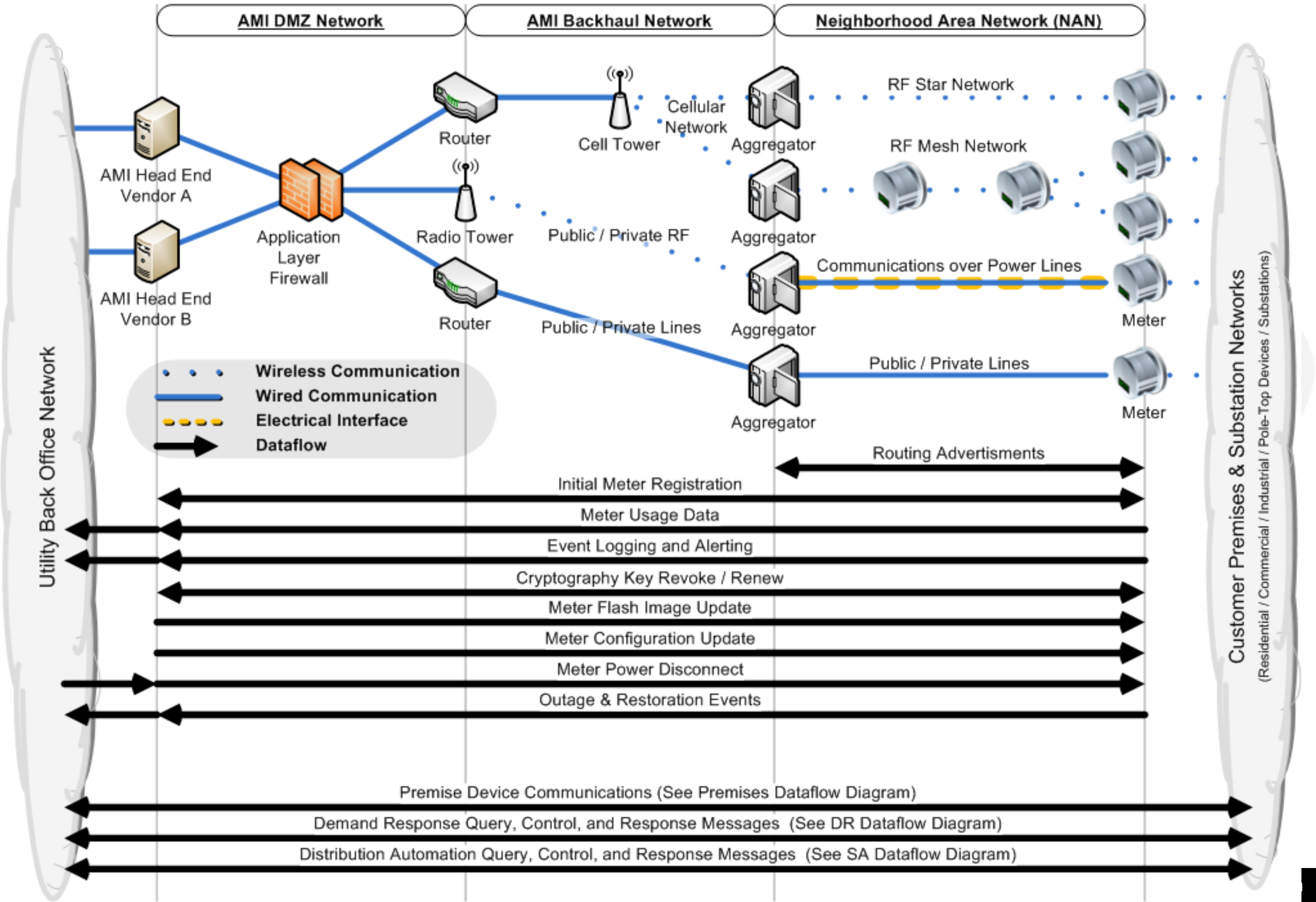
Application Layer (OSI 5-7):
- ANSI C12.22
- ANSI C12.18 / C12.19 / C12.21

Transport Layer (OSI 4):
- TCP / UDP
- ANSI C12.22

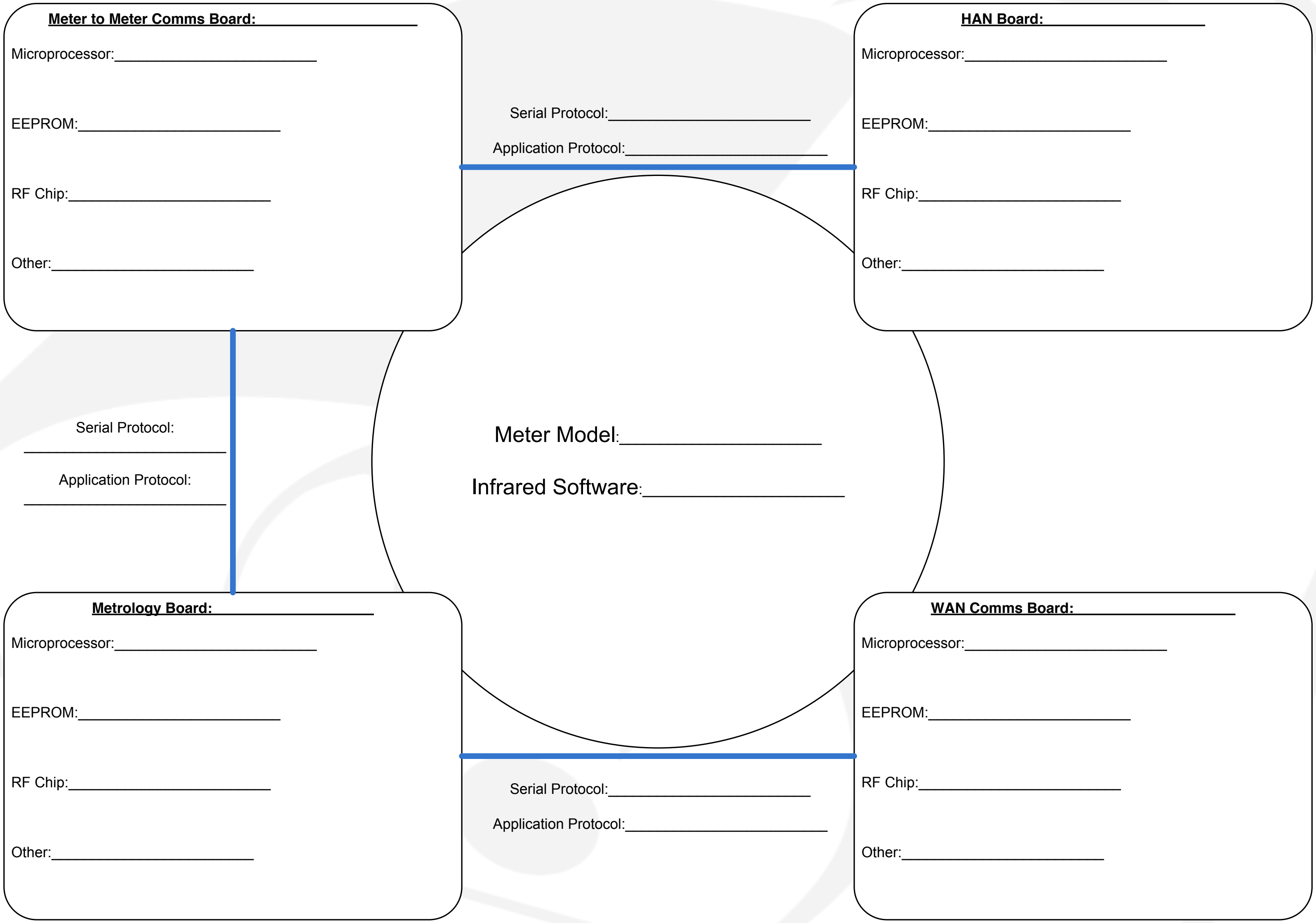
Network Layer (OSI 3):
- IP4 / IP6
- ANSI c12.22

PHY / MAC Layter (OSI 1-2):
- Radio (WiMax 802.16d/e)
- Proprietary Radio (900 MHz)
- Cellular (EVDO / GPRS / …)
- Power Line Carrier (BPL / P1901)
- Fiber (RFoG / FTTP / Ethernet)

AMI Dataflow Diagram


AMI DMZ Network
AMI Backhaul Network
Neighborhood Area Network (NAN)
Utility Back Office Network
AMI Head End Vendor A
AMI Head End Vendor B
Application Layer Firewall
Router
Radio Tower
Router
Cell Tower
Cellular Network
Public / Private RF
Public / Private Lines
Aggregator
Aggregator
Aggregator
Aggregator
RF Star Network
RF Mesh Network
Communications over Power Lines
Public / Private Lines
Meter
Meter
Customer Premises & Substation Networks
(Residential / Commercial / Industrial / Pole-Top Devices / Substations)
Wireless Communication
Wired Communication
Electrical Interface
Dataflow
Routing Advertisments
Initial Meter Registration
Meter Usage Data
Event Logging and Alerting
Cryptography Key Revoke / Renew
Meter Flash Image Update
Meter Configuration Update
Meter Power Disconnect
Outage & Restoration Events
Premise Device Communications (See Premises Dataflow Diagram)
Demand Response Query, Control, and Response Messages (See DR Dataflow Diagram)
Distribution Automation Query, Control, and Response Messages (See SA Dataflow Diagram)

AMI Meter Comonents



Meter to Meter Comms Board: ____________________

Microprocessor: ____________________

EEPROM: ____________________

RF Chip: ____________________

Other: ____________________

Serial Protocol: ____________________

Application Protocol: ____________________

HAN Board: ____________________

Microprocessor: ____________________

EEPROM: ____________________

RF Chip: ____________________

Other: ____________________

Serial Protocol:

____

Application Protocol:

____

Meter Model: ____________________

Infrared Software: ____________________

Metrology Board: ____________________

Microprocessor: ____________________

EEPROM: ____________________

RF Chip: ____________________

Other: ____________________

Serial Protocol: ____________________

Application Protocol: ____________________

WAN Comms Board: ____________________

Microprocessor: ____________________

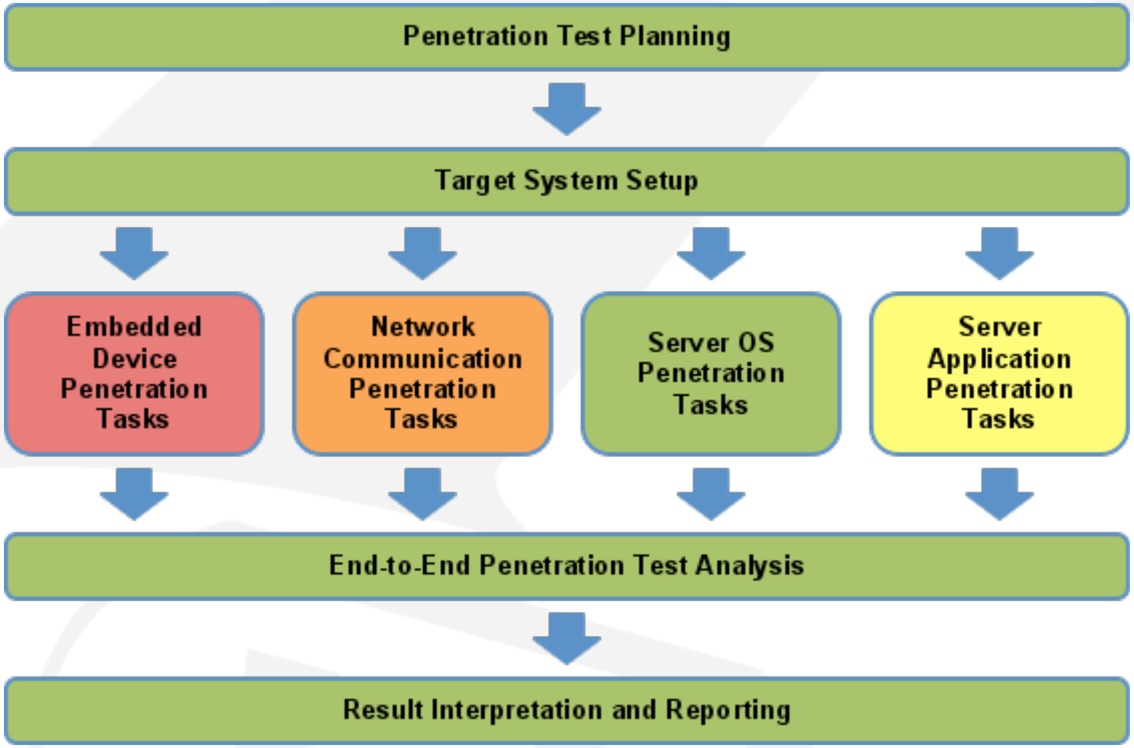
EEPROM: ____________________

RF Chip: ____________________

Other: ____________________

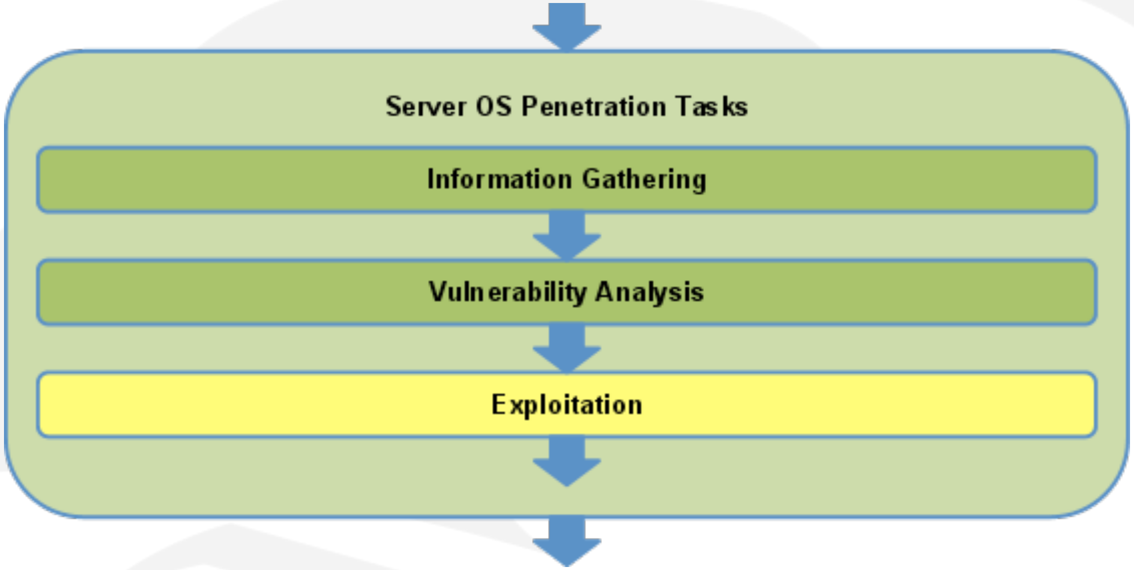
Penetration Testing Methodology

Smart Grid Penetration Test Plan


Penetration Test Planning
Target System Setup
Embedded Device Penetration Tasks
Network Communication Penetration Tasks
Server OS Penetration Tasks
Server Application Penetration Tasks
End-to-End Penetration Test Analysis
Result Interpretation and Reporting

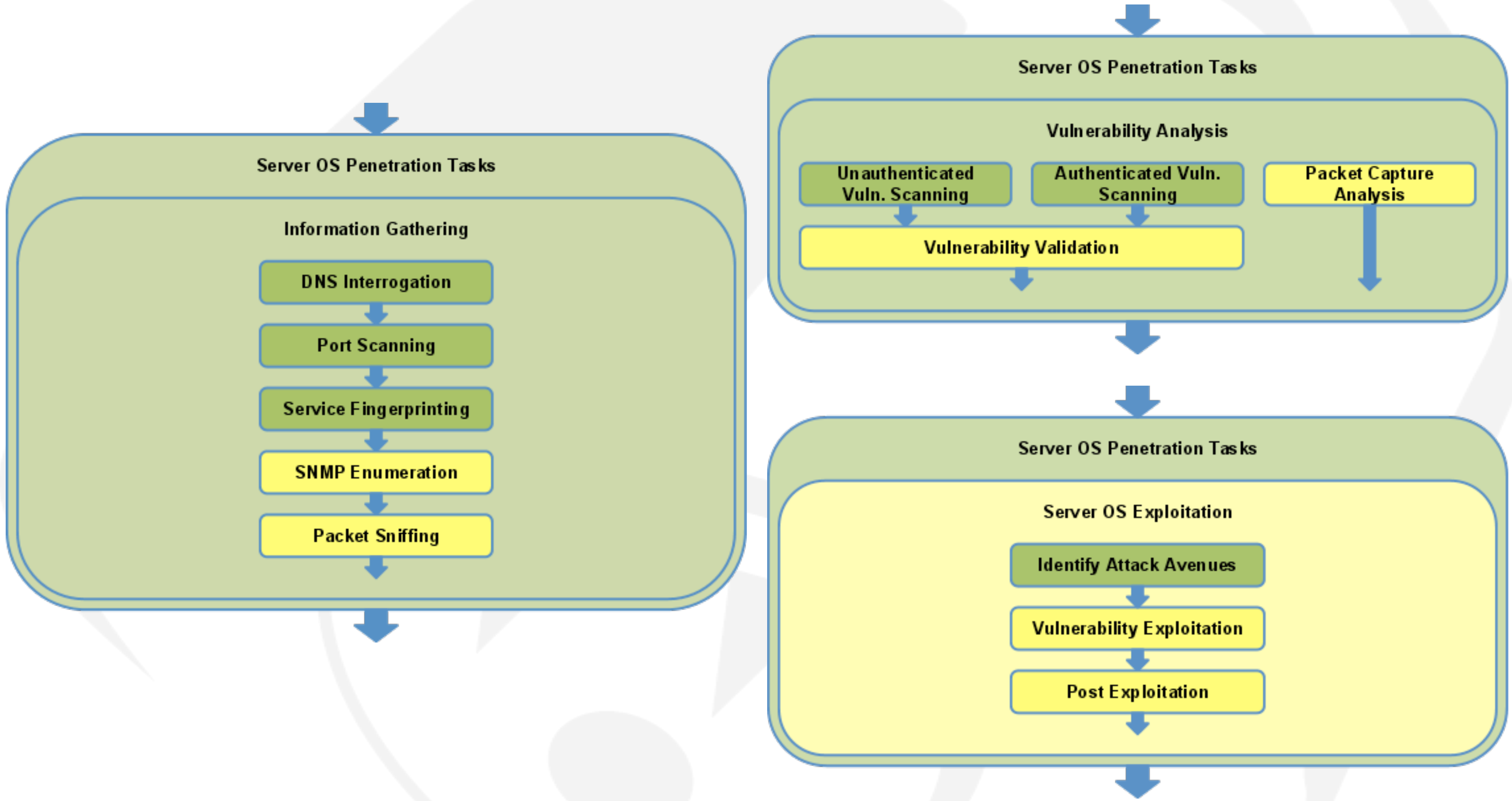
- Green: Tasks most frequently and require the most basic of penetration testing skill
- Yellow: Tasks commonly performed and require moderate penetration testing skill
- Orange: Tasks that are occasionally performed but require higher levels of expertise
- Red: Tasks performed infrequently and require highly specialized skills

Server OS Task Sub-Categories



- Most servers in the datacenter controlling Smart Grid systems are running commodity OSes like Windows and Linux
- Skills needed to pentest these systems are no different than non-smart grid pentests
- Level of care when testing production systems is greatly increased
- Mastery and understanding of automated tools used is critical

Server OS Pentest Tasks



Server OS Penetration Tasks

Information Gathering

DNS Interrogation

Port Scanning

Service Fingerprinting

SNMP Enumeration

Packet Sniffing

Server OS Penetration Tasks

Vulnerability Analysis

Unauthenticated Vuln. Scanning

Authenticated Vuln. Scanning

Packet Capture Analysis

Vulnerability Validation

Server OS Penetration Tasks

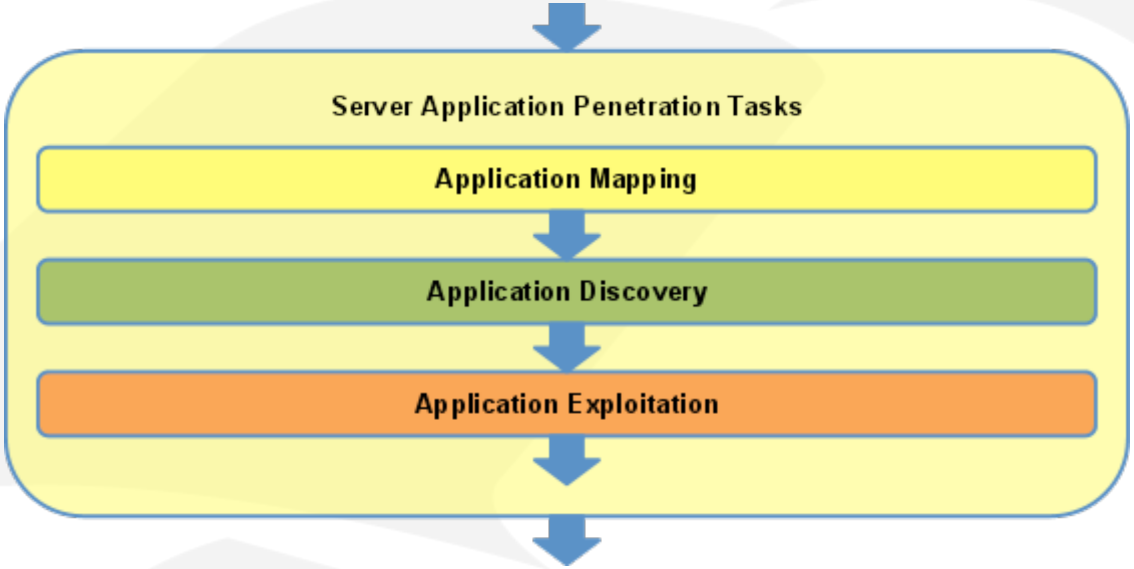
Server OS Exploitation

Identify Attack Avenues

Vulnerability Exploitation

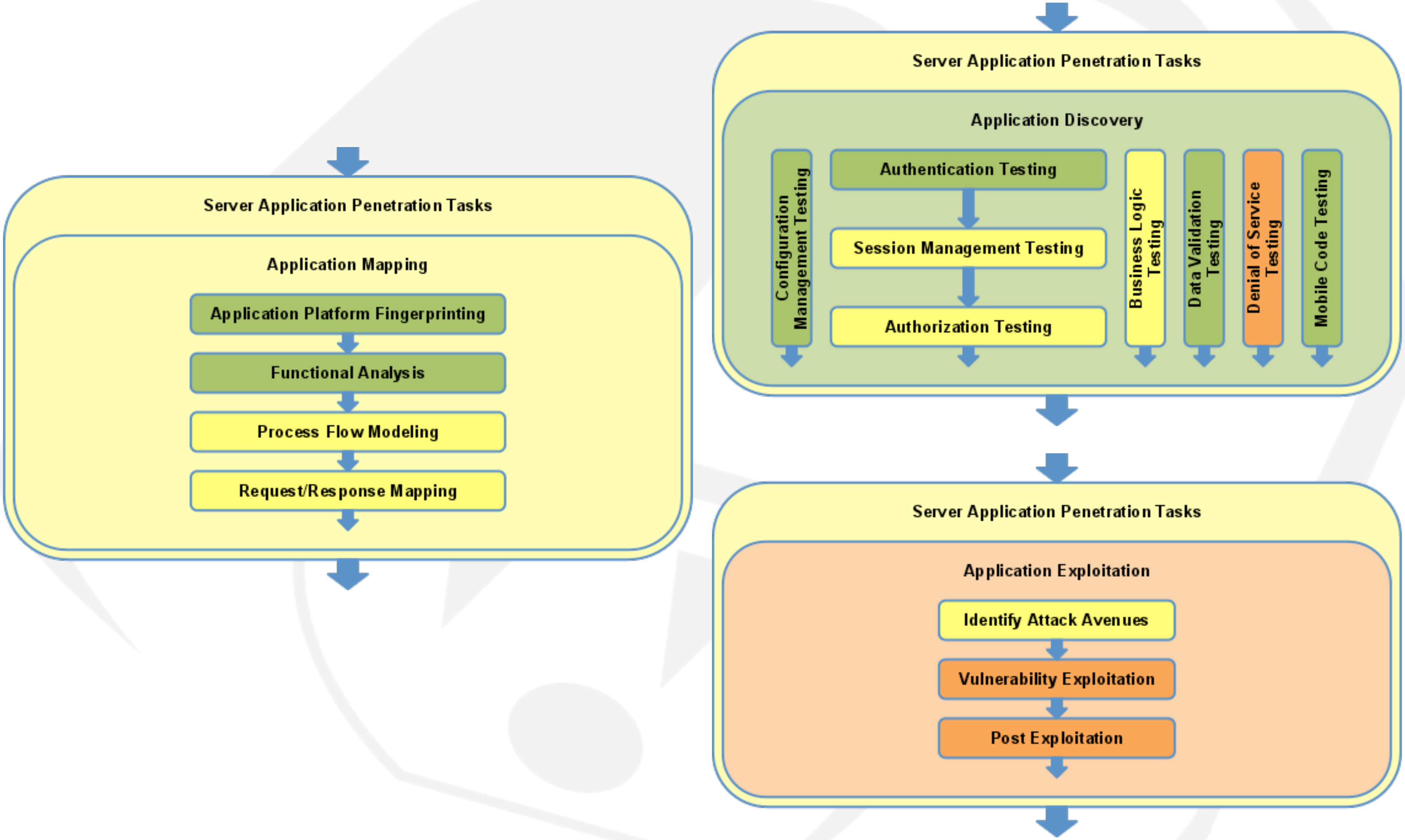
Post Exploitation

Server App Task Sub-Categories



- Includes all user interfaces and smart grid services
- Most modern user interfaces are web applications or fat applications speaking to a web service
- Most server-to-server communications use SOAP or REST web services, but other interfaces like RPC are occasionally seen
- Automated tools can be VERY dangerous in these applications as POST reqests can shut down power or brick field devices

Server App Pentest Tasks



Server Application Penetration Tasks

Application Mapping

- Application Platform Fingerprinting
- Functional Analysis
- Process Flow Modeling
- Request/Response Mapping

Server Application Penetration Tasks

Application Discovery

- Configuration Management Testing
- Authentication Testing
- Session Management Testing
- Authorization Testing
- Business Logic Testing
- Data Validation Testing
- Denial of Service Testing
- Mobile Code Testing

Server Application Penetration Tasks

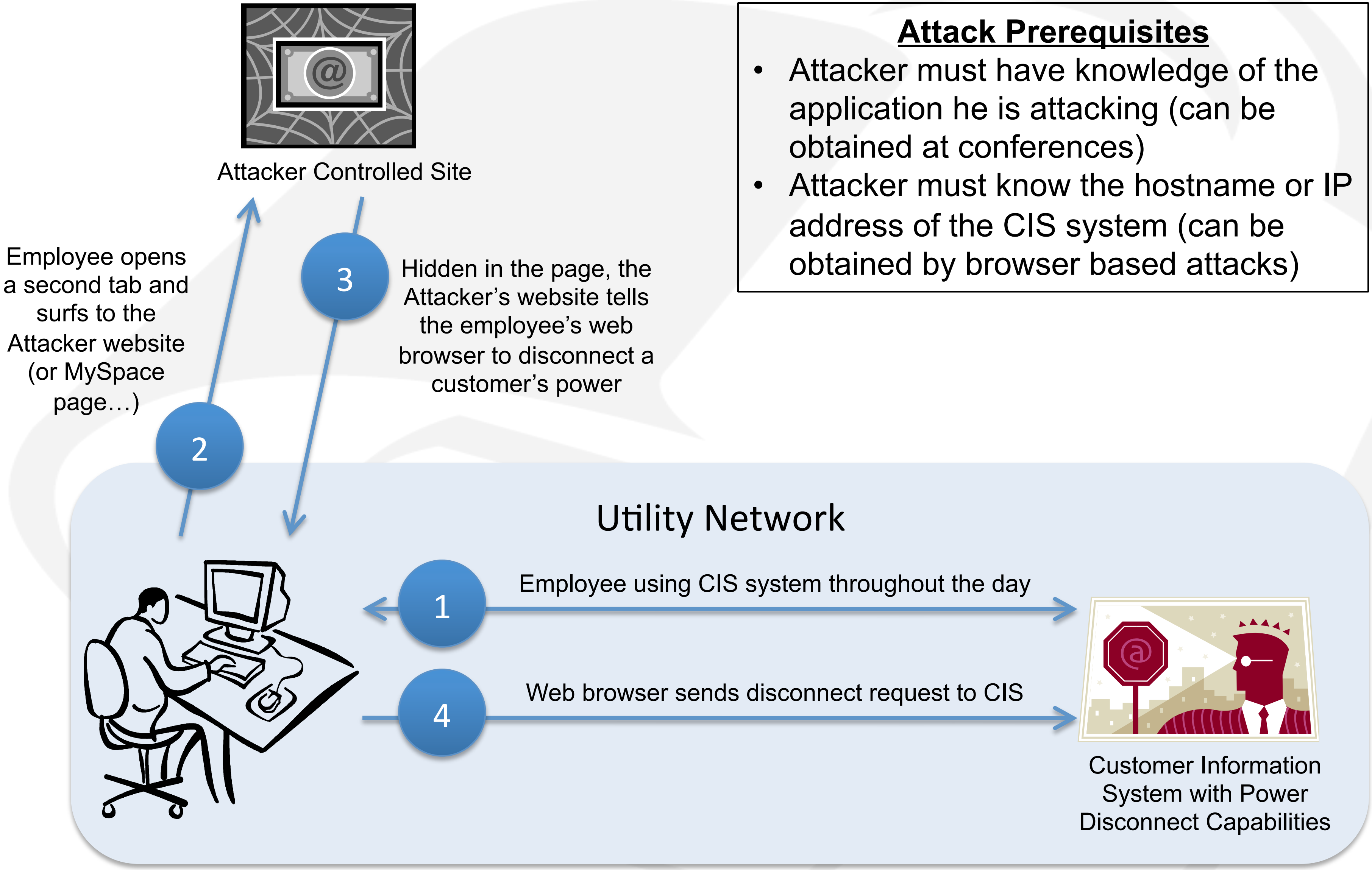
Application Exploitation

- Identify Attack Avenues
- Vulnerability Exploitation
- Post Exploitation

Task: Session Management (CSRF)

Attack Prerequisites

- Attacker must have knowledge of the application he is attacking (can be obtained at conferences)
- Attacker must know the hostname or IP address of the CIS system (can be obtained by browser based attacks)



Attacker Controlled Site

2 Employee opens a second tab and surfs to the Attacker website (or MySpace page…)

3 Hidden in the page, the Attacker's website tells the employee's web browser to disconnect a customer's power

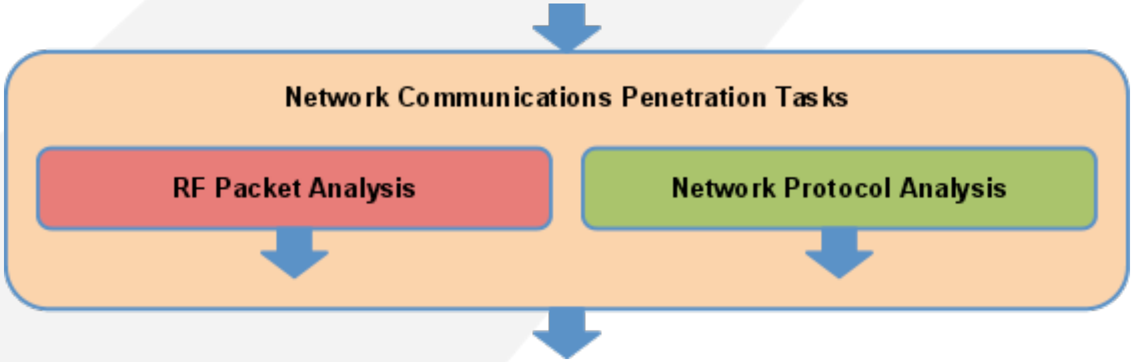
Utility Network

1 Employee using CIS system throughout the day

4 Web browser sends disconnect request to CIS

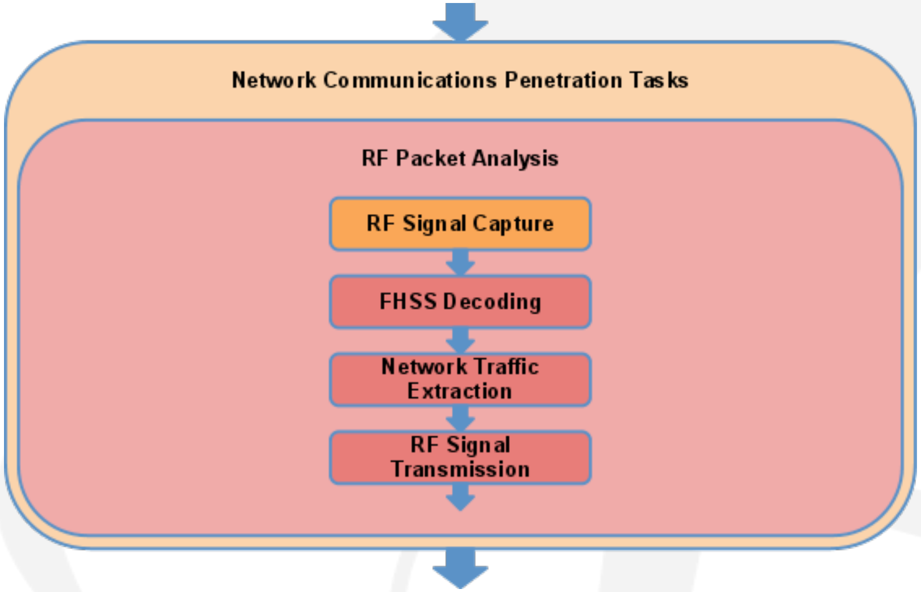
Customer Information System with Power Disconnect Capabilities

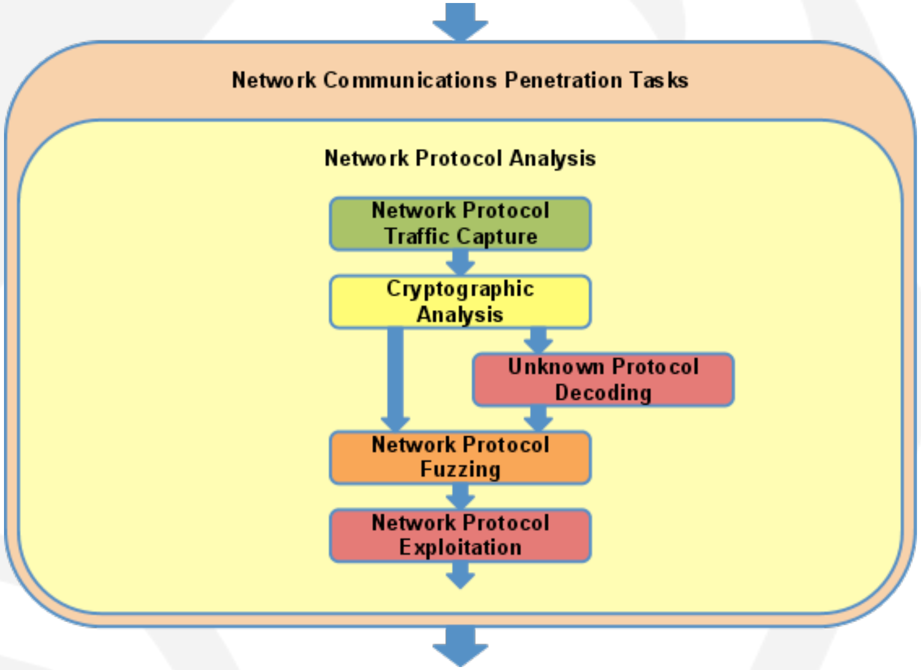
Network Coms Task Sub-Categories


Network Communications Penetration Tasks
RF Packet Analysis
Network Protocol Analysis

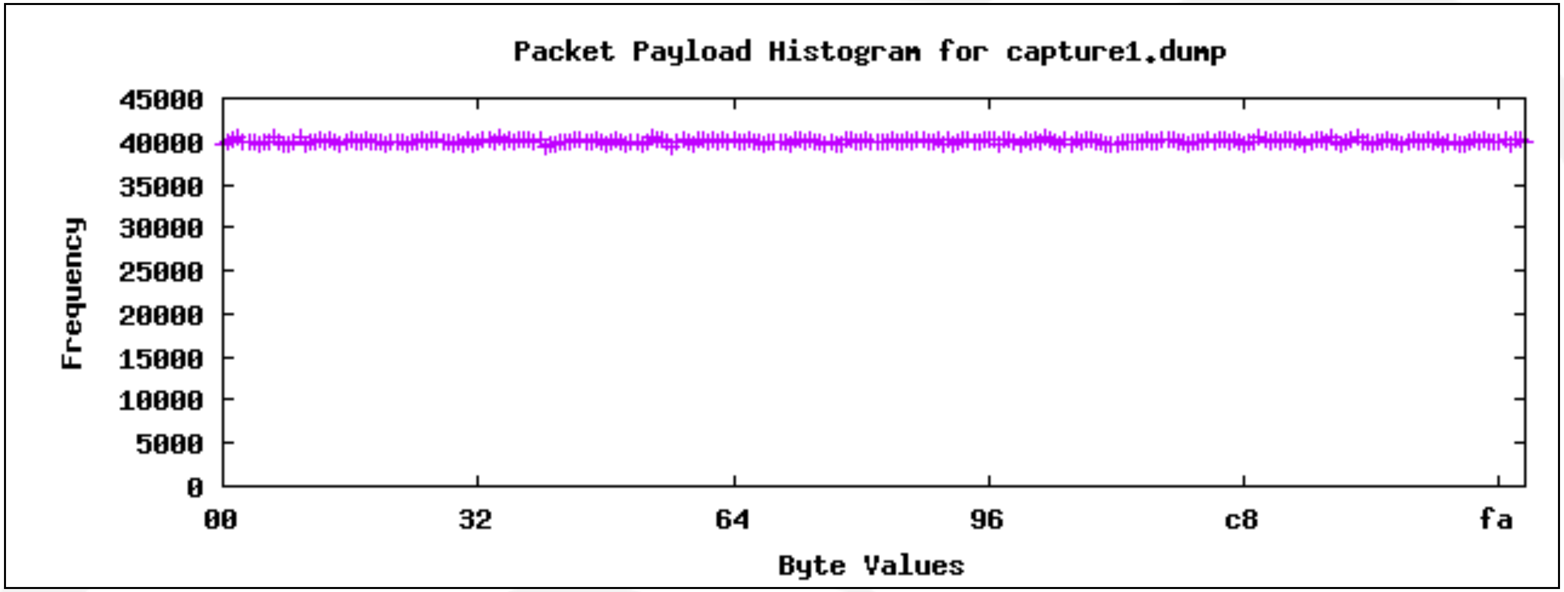
- These tasks include all network traffic between any of the devices regardless of its location such as sever-to-server, server-to-device, device-to-device
- RF Packet Analysis is not commonly performed because we assume all security is handled in the network protocols
  - Frequency hopping isn't a security control

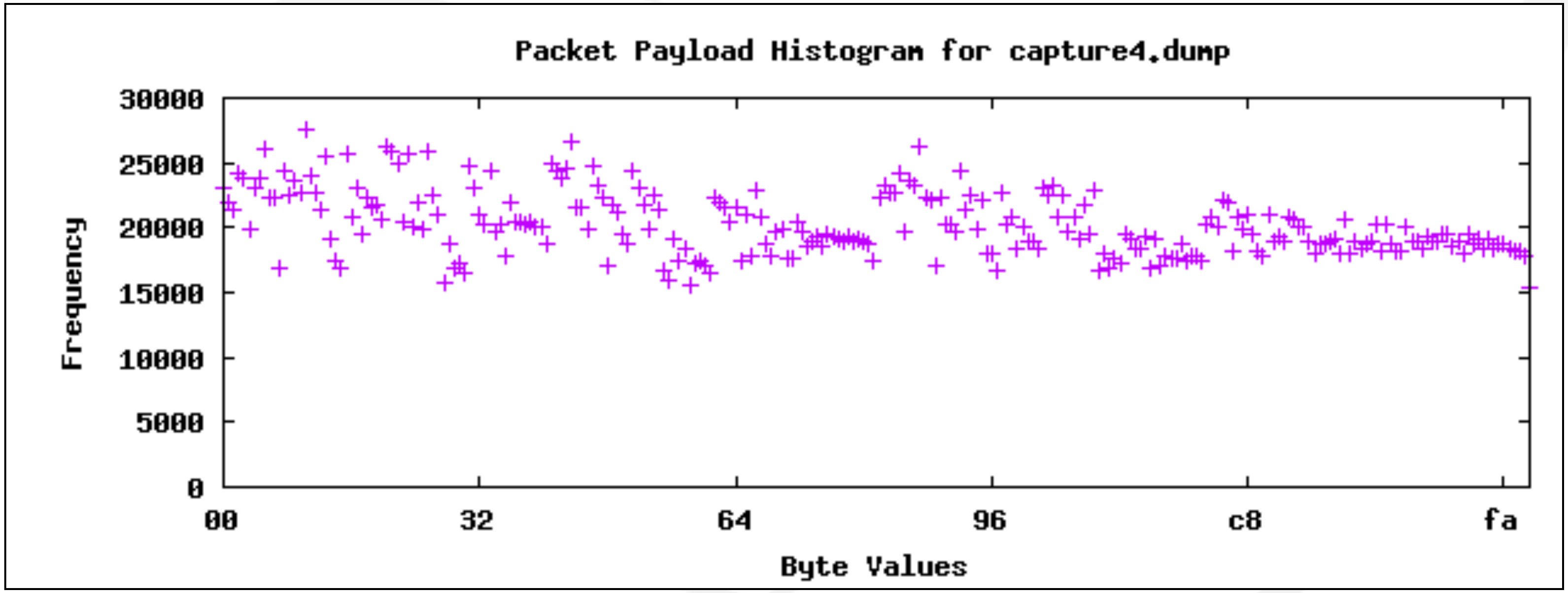
Network Coms Pentest Tasks




Network Communications Penetration Tasks
Network Protocol Analysis
Network Protocol Traffic Capture
Cryptographic Analysis
Unknown Protocol Decoding
Network Protocol Fuzzing
Network Protocol Exploitation

Task: Cryptographic Analysis



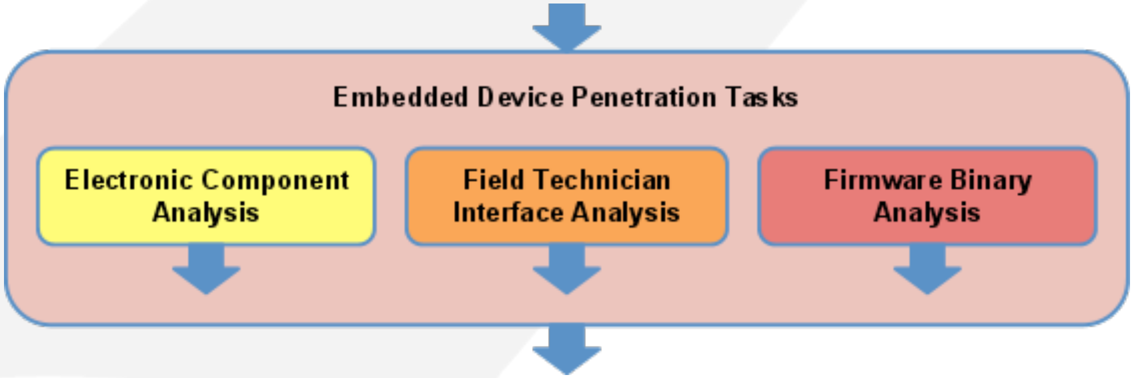


Insecure Block Cipher Modes

- AES ciphers using CTR mode effectively become a stream cipher
- Without key derivation and rotation, IV collisions compromise integrity of cipher

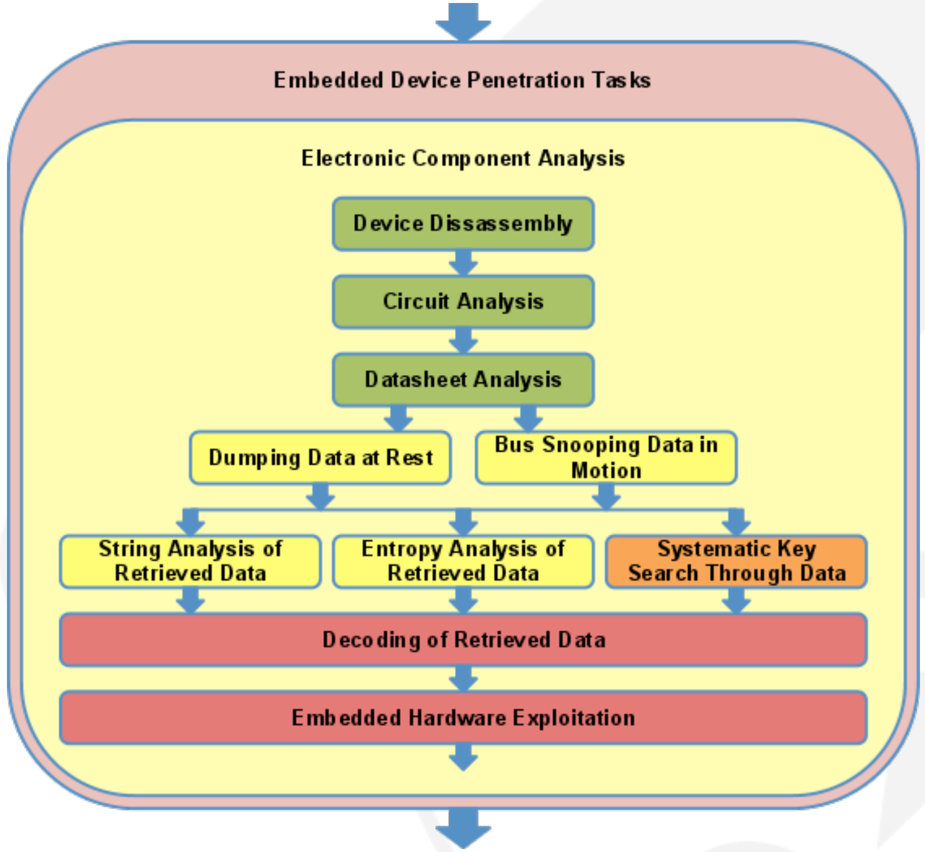
```
C:\>type ivcoltest.py
#!/usr/bin/env python
knownplain = "\xaa\xaa\x03\x00\x00\x00\x08\x00\x45\x00\x01\x48\x00\x01\x00\x00"
knowncip = "\x31\xb9\x84\x81\xe1\x96\x6e\x71\xd8\xa3\x39\x0c\xfb\x48\xaa\x61"
unknowncip = "\x31\xb9\x84\x81\xe1\x96\x6e\x71\xd8\xa3\x3d\x0c\xfb\xb5\xaa\x61"
print "Decrypted packet: "
for i in range(0,len(knownplain)):
    print "%02x"%( (ord(knownplain[i]) ^ ord(knowncip[i])) ^ ord(unknowncip[i]) ),
print("\n")

C:\>python ivcoltest.py
Decrypted packet:
aa aa 03 00 00 00 08 00 45 00 05 48 00 fc 00 00
```

Embedded Task Sub-Categories


Embedded Device Penetration Tasks
Electronic Component Analysis
Field Technician Interface Analysis
Firmware Binary Analysis

- These tasks target physical attacks on embedded field devices:
  - electronic components that store data (EEPROM, Flash, RAM, MCu storage)
  - buses that pass data between components (parallel buses and serial buses)
  - input interfaces used for administrative or debugging purposes (serial ports, parallel ports, infrared/optical ports)
- Overarching goal for embedded device testing is to identify vulnerabilities that allow attackers to expand their control of that single device to other devices with limited or no physical access to those other devices

Embedded Device Pentest Tasks



Embedded Device Penetration Tasks

Electronic Component Analysis

Device Dissassembly

Circuit Analysis

Datasheet Analysis

Dumping Data at Rest

Bus Snooping Data in Motion

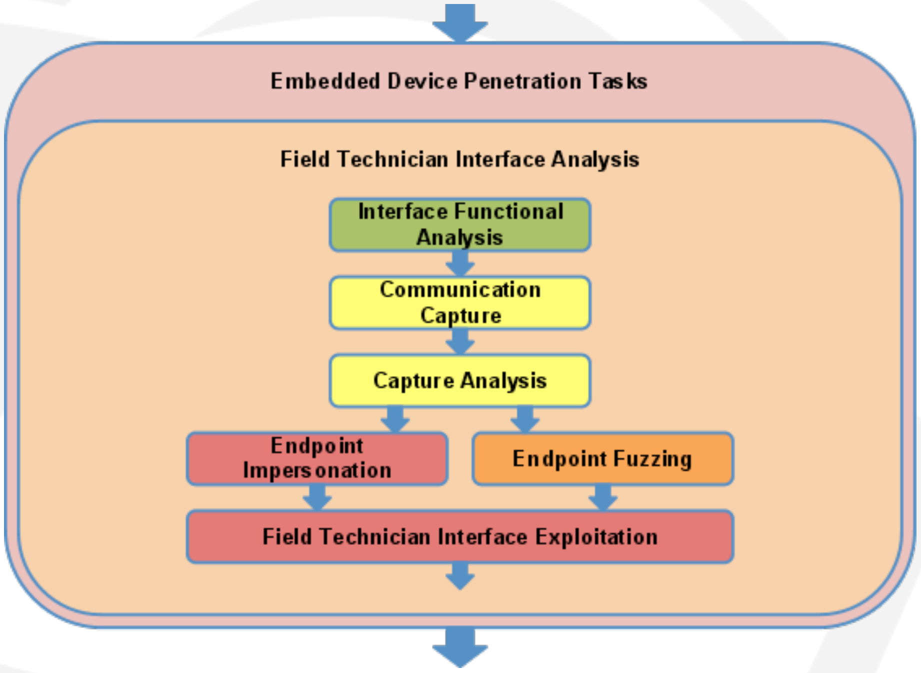
String Analysis of Retrieved Data

Entropy Analysis of Retrieved Data

Systematic Key Search Through Data

Decoding of Retrieved Data

Embedded Hardware Exploitation



Embedded Device Penetration Tasks

Field Technician Interface Analysis

Interface Functional Analysis

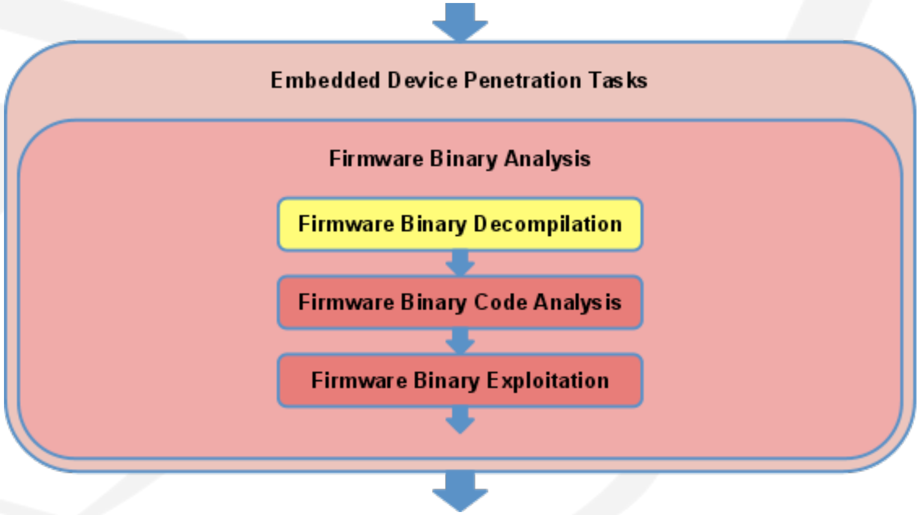
Communication Capture

Capture Analysis

Endpoint Impersonation

Endpoint Fuzzing

Field Technician Interface Exploitation



Embedded Device Penetration Tasks

Firmware Binary Analysis

Firmware Binary Decompilation

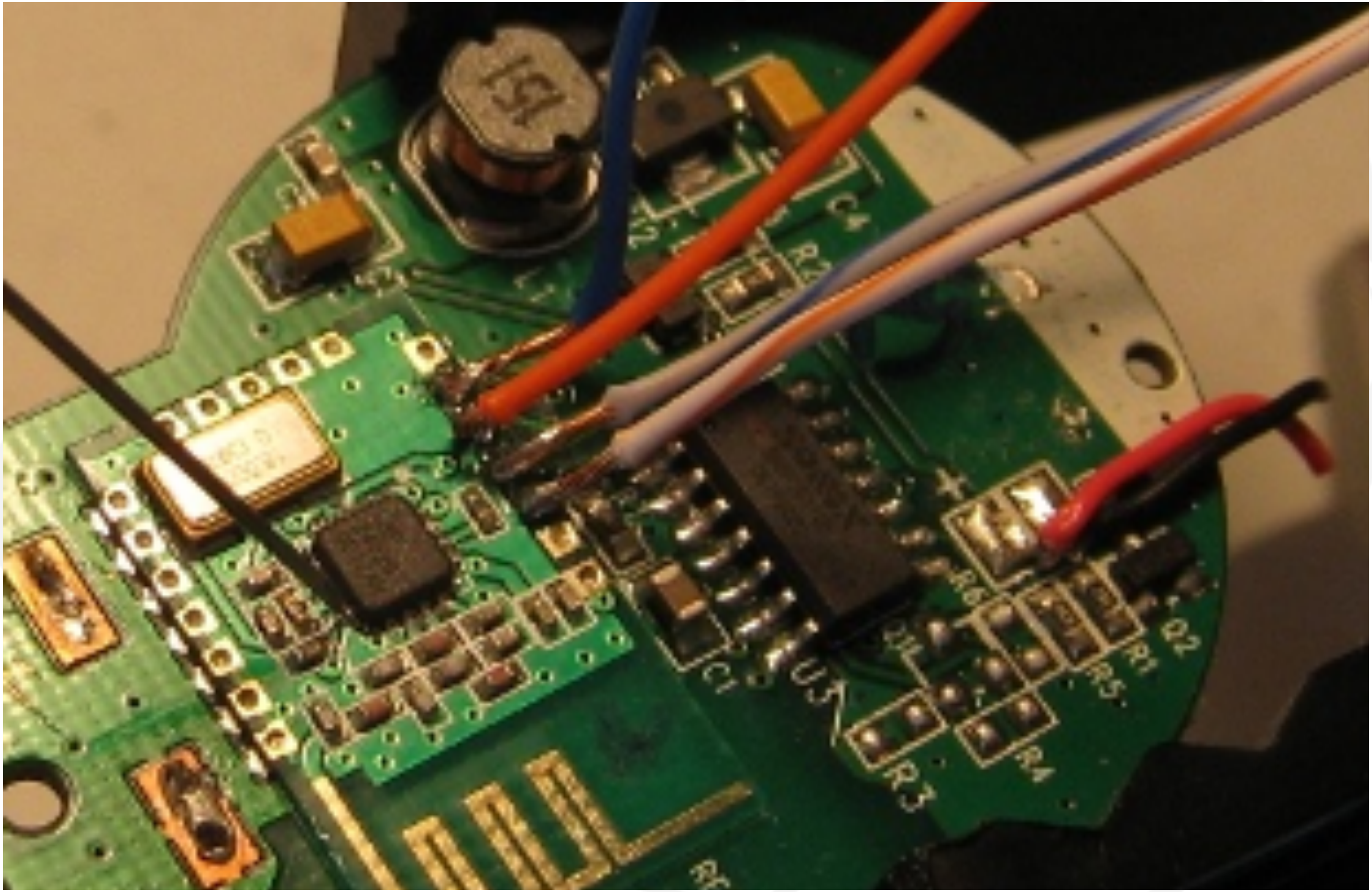
Firmware Binary Code Analysis

Firmware Binary Exploitation

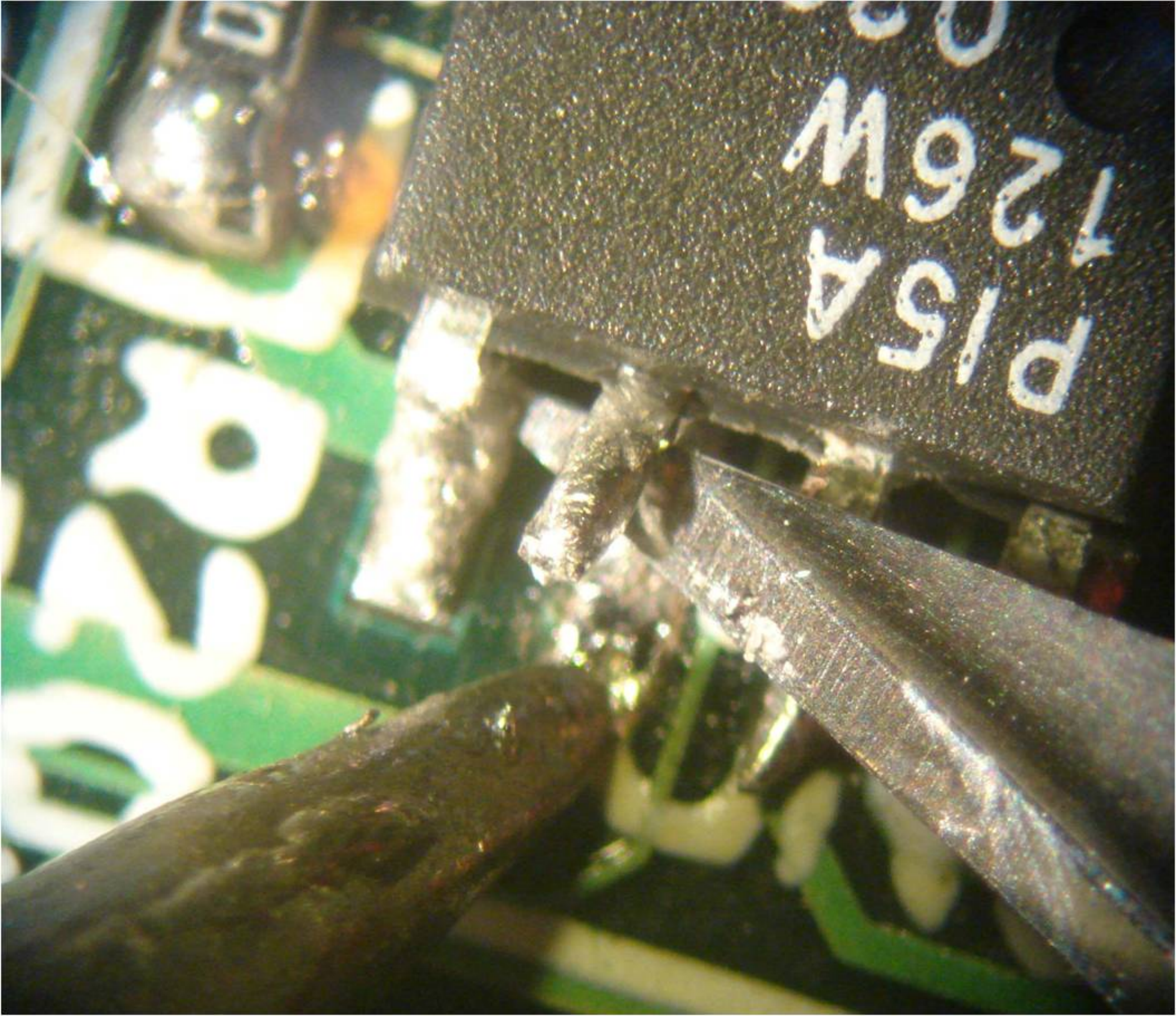
Goals: Key & Firmware Extraction

- Attacking data at rest
  - Power down the device, expose its circuit board, and interact directly with each component
  - Extract contents of accessible RAM, Flash, and EEPROM
  - Identify cryptography keys or firmware
- Attacking data in motion
  - Boot and normally operate the device in a lab, monitoring bus activity between major chips (MCU, Radio, Flash, RAM)
  - Crypto keys can often be found in key load operations between a microcontroller and crypto accelerator
  - Firmware can often be found in boot processes (between Flash and MCU) and firmware updates (between Radio, MCU, and Flash)

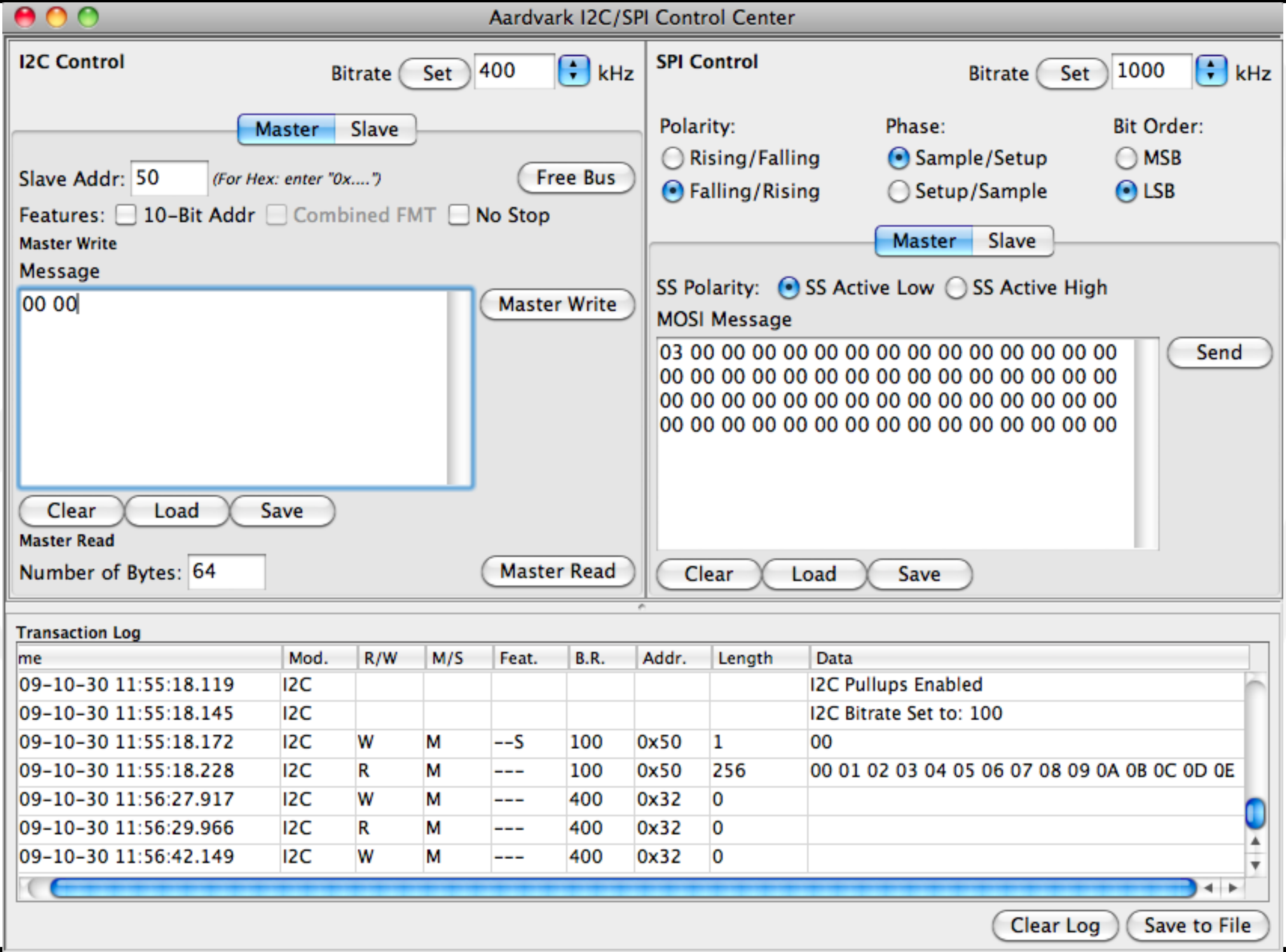
Interfacing with an IC



Lifting an IC's Chip Enable (CE) Pin



Task: Dumping Data at Rest



Aardvark I2C/SPI Control Center

I2C Control

Bitrate 400 kHz

Slave Addr: 50 *(For Hex: enter "0x....")*

Features: □ 10-Bit Addr □ Combined FMT □ No Stop

Master Write

Message

```
00 00
```

Master Read

Number of Bytes: 64

SPI Control

Bitrate 1000 kHz

Polarity: ○ Rising/Falling ● Falling/Rising

Phase: ● Sample/Setup ○ Setup/Sample

Bit Order: ○ MSB ● LSB

SS Polarity: ● SS Active Low ○ SS Active High

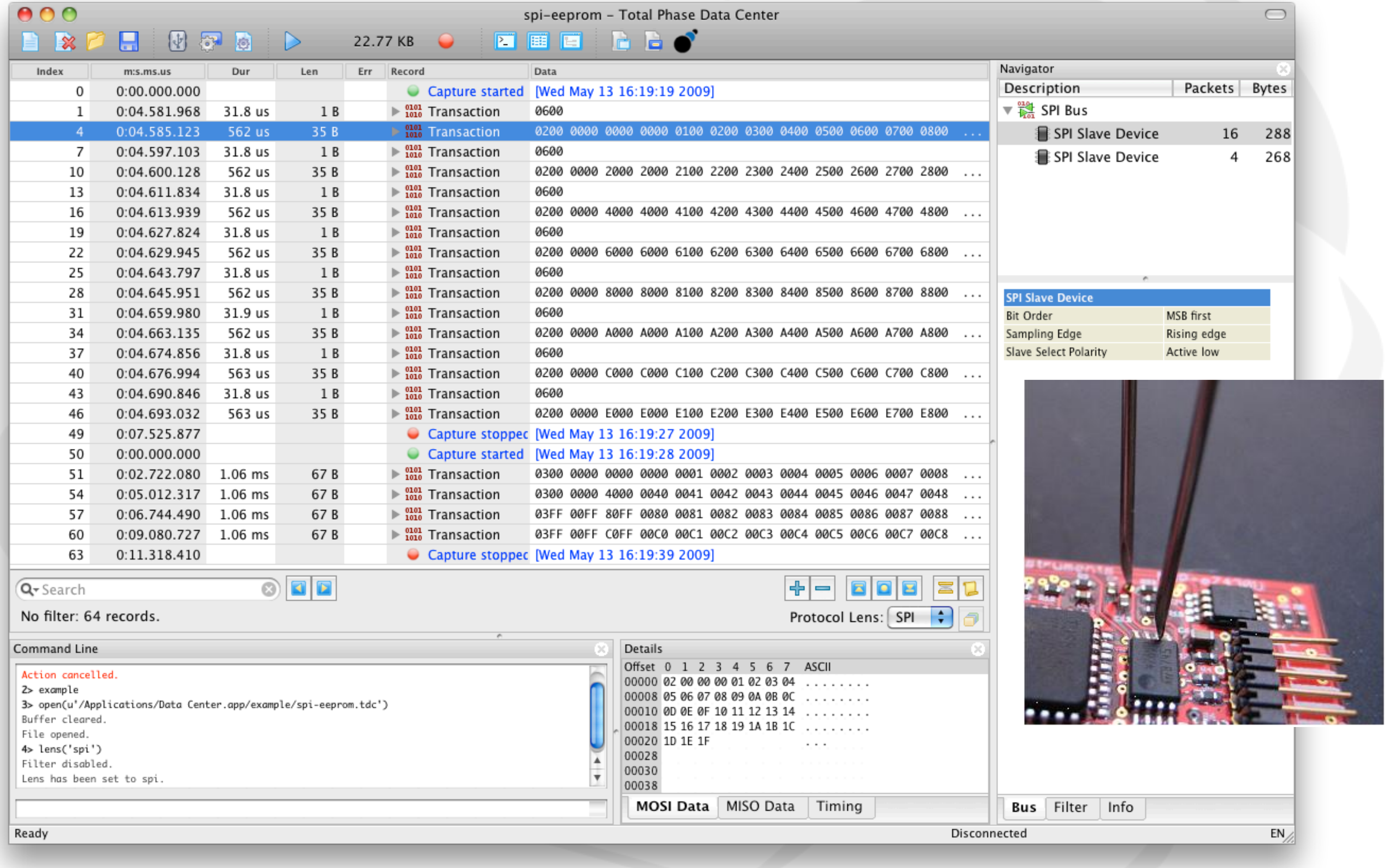
MOSI Message

```
03 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Transaction Log

me	Mod.	R/W	M/S	Feat.	B.R.	Addr.	Length	Data
09-10-30 11:55:18.119	I2C							I2C Pullups Enabled
09-10-30 11:55:18.145	I2C							I2C Bitrate Set to: 100
09-10-30 11:55:18.172	I2C	W	M	--S	100	0x50	1	00
09-10-30 11:55:18.228	I2C	R	M	---	100	0x50	256	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E
09-10-30 11:56:27.917	I2C	W	M	---	400	0x32	0	
09-10-30 11:56:29.966	I2C	R	M	---	400	0x32	0	
09-10-30 11:56:42.149	I2C	W	M	---	400	0x32	0	

Task: Bus Snooping Data in Motion



spi-eeprom – Total Phase Data Center

Index	m:s.ms.us	Dur	Len	Err	Record	Data
0	0:00.000.000				● Capture started	[Wed May 13 16:19:19 2009]
1	0:04.581.968	31.8 us	1 B		Transaction	0600
4	0:04.585.123	562 us	35 B		Transaction	0200 0000 0000 0000 0100 0200 0300 0400 0500 0600 0700 0800 ...
7	0:04.597.103	31.8 us	1 B		Transaction	0600
10	0:04.600.128	562 us	35 B		Transaction	0200 0000 2000 2000 2100 2200 2300 2400 2500 2600 2700 2800 ...
13	0:04.611.834	31.8 us	1 B		Transaction	0600
16	0:04.613.939	562 us	35 B		Transaction	0200 0000 4000 4000 4100 4200 4300 4400 4500 4600 4700 4800 ...
19	0:04.627.824	31.8 us	1 B		Transaction	0600
22	0:04.629.945	562 us	35 B		Transaction	0200 0000 6000 6000 6100 6200 6300 6400 6500 6600 6700 6800 ...
25	0:04.643.797	31.8 us	1 B		Transaction	0600
28	0:04.645.951	562 us	35 B		Transaction	0200 0000 8000 8000 8100 8200 8300 8400 8500 8600 8700 8800 ...
31	0:04.659.980	31.9 us	1 B		Transaction	0600
34	0:04.663.135	562 us	35 B		Transaction	0200 0000 A000 A000 A100 A200 A300 A400 A500 A600 A700 A800 ...
37	0:04.674.856	31.8 us	1 B		Transaction	0600
40	0:04.676.994	563 us	35 B		Transaction	0200 0000 C000 C000 C100 C200 C300 C400 C500 C600 C700 C800 ...
43	0:04.690.846	31.8 us	1 B		Transaction	0600
46	0:04.693.032	563 us	35 B		Transaction	0200 0000 E000 E000 E100 E200 E300 E400 E500 E600 E700 E800 ...
49	0:07.525.877				● Capture stoppec	[Wed May 13 16:19:27 2009]
50	0:00.000.000				● Capture started	[Wed May 13 16:19:28 2009]
51	0:02.722.080	1.06 ms	67 B		Transaction	0300 0000 0000 0000 0001 0002 0003 0004 0005 0006 0007 0008 ...
54	0:05.012.317	1.06 ms	67 B		Transaction	0300 0000 4000 0040 0041 0042 0043 0044 0045 0046 0047 0048 ...
57	0:06.744.490	1.06 ms	67 B		Transaction	03FF 00FF 80FF 0080 0081 0082 0083 0084 0085 0086 0087 0088 ...
60	0:09.080.727	1.06 ms	67 B		Transaction	03FF 00FF C0FF 00C0 00C1 00C2 00C3 00C4 00C5 00C6 00C7 00C8 ...
63	0:11.318.410				● Capture stoppec	[Wed May 13 16:19:39 2009]

No filter: 64 records.

Protocol Lens: SPI

Navigator

Description	Packets	Bytes
SPI Bus		
SPI Slave Device	16	288
SPI Slave Device	4	268

SPI Slave Device

Bit Order	MSB first
Sampling Edge	Rising edge
Slave Select Polarity	Active low

Command Line

```
Action cancelled.
2> example
3> open(u'/Applications/Data Center.app/example/spi-eeprom.tdc')
Buffer cleared.
File opened.
4> lens('spi')
Filter disabled.
Lens has been set to spi.
```

Details

```
Offset 0  1  2  3  4  5  6  7   ASCII
00000 02 00 00 00 01 02 03 04  ........
00008 05 06 07 08 09 0A 0B 0C  ........
00010 0D 0E 0F 10 11 12 13 14  ........
00018 15 16 17 18 19 1A 1B 1C  ........
00020 1D 1E 1F                 ...
00028
00030
00038
```

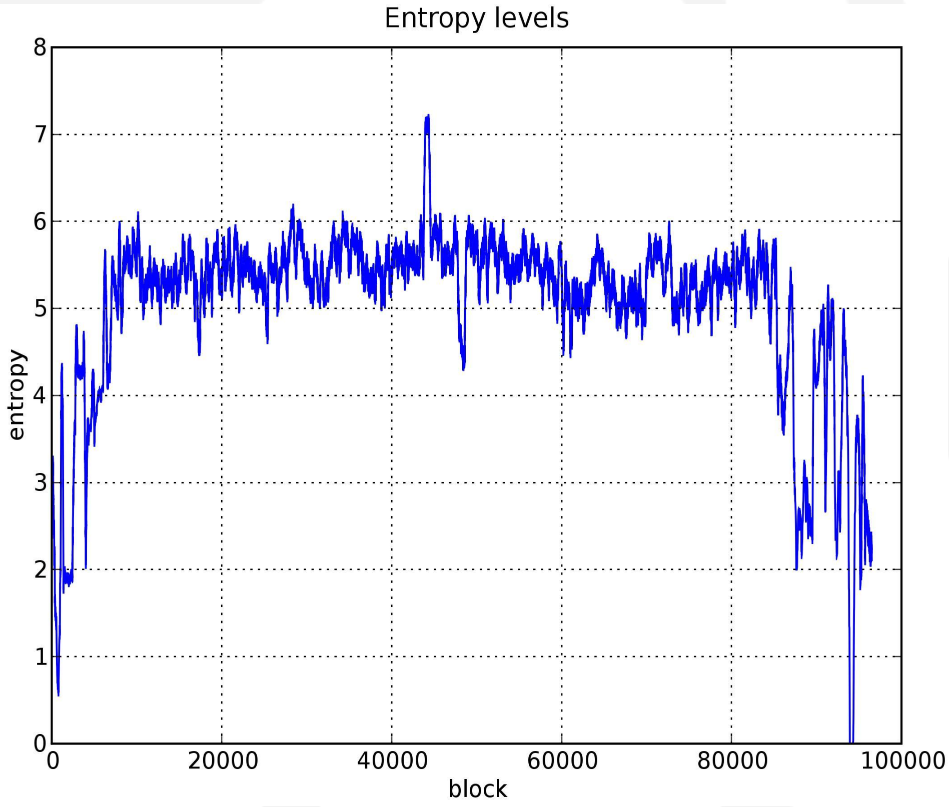
Task: Systematic Key Search

- Perform basic string searches for obvious keys
- Develop custom tools to do more advanced searches:
  - GoodFET: Abuses vulnerability in TI, Ember radios to access RAM even when chip is locked
  - zbgoodfind: Search for ZigBee key using RAM dump as a list of potential keys
  - Combined they can recover the ZigBee network key

```
$ sudo goodfet.cc dumpdata chipcon-2430-mem.hex
Target identifies as CC2430/r04.
Dumping data from e000 to ffff as chipcon-2430-mem.hex.
...
$ objcopy -I ihex -O binary chipcon-2430-mem.hex chipcon-2430-mem.bin
$ zbgoodfind -R encdata.dcf -f chipcon-2430-mem.hex
zbgoodfind: searching the contents of chipcon-2430-mem.hex for
encryption keys with the first encrypted packet in encdata.dcf.
Key found after 6397 guesses:  c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc
cd ce cf
```

Task: Entropy Analysis of Data

- Asymmetric keys have high entropy (very random)
- RAM and Flash is filled with non-random data
- Graphing entropy of flash reveals a spike in randomness
- This spike is the location of the asymmetric key in flash



SamuraiSTFU

- SamuraiSTFU (Security Testing Framework for Utilities)
- Leverage last 5 years of experience developing and managing the SamuraiWTF (Web Testing Framework) project
- Live DVD / VM for Smart Grid penetration testing
  - Primary audience is electric utility security teams
  - Secondary audience is security contractors and independent researchers
- Include "cream of the crop" free and open source tools for all aspects of SG Pentesting
  - Best web pentesting tools (small subset of SamuraiWTF)
  - Best network pentesting tools (small subset of Backtrack)
  - Best hardware pentesting tools (not currently included on any distribution)
- Include documentation on tools, architecture, methodology, and protocols
- Includes simulators, sample packet captures, and data dumps
- Will launch soon at www.SamuraiSTFU.org

Contact Information

www.utilisec.com
sales@utilisec.com

Justin Searle
personal: justin@meeas.com
work: justin@utilisec.com
cell: 801-784-2052
twitter: @meeas