



Attacking and Defending the Grid

Pulling back the curtains to reveal the front battle lines of Smart Grid security.

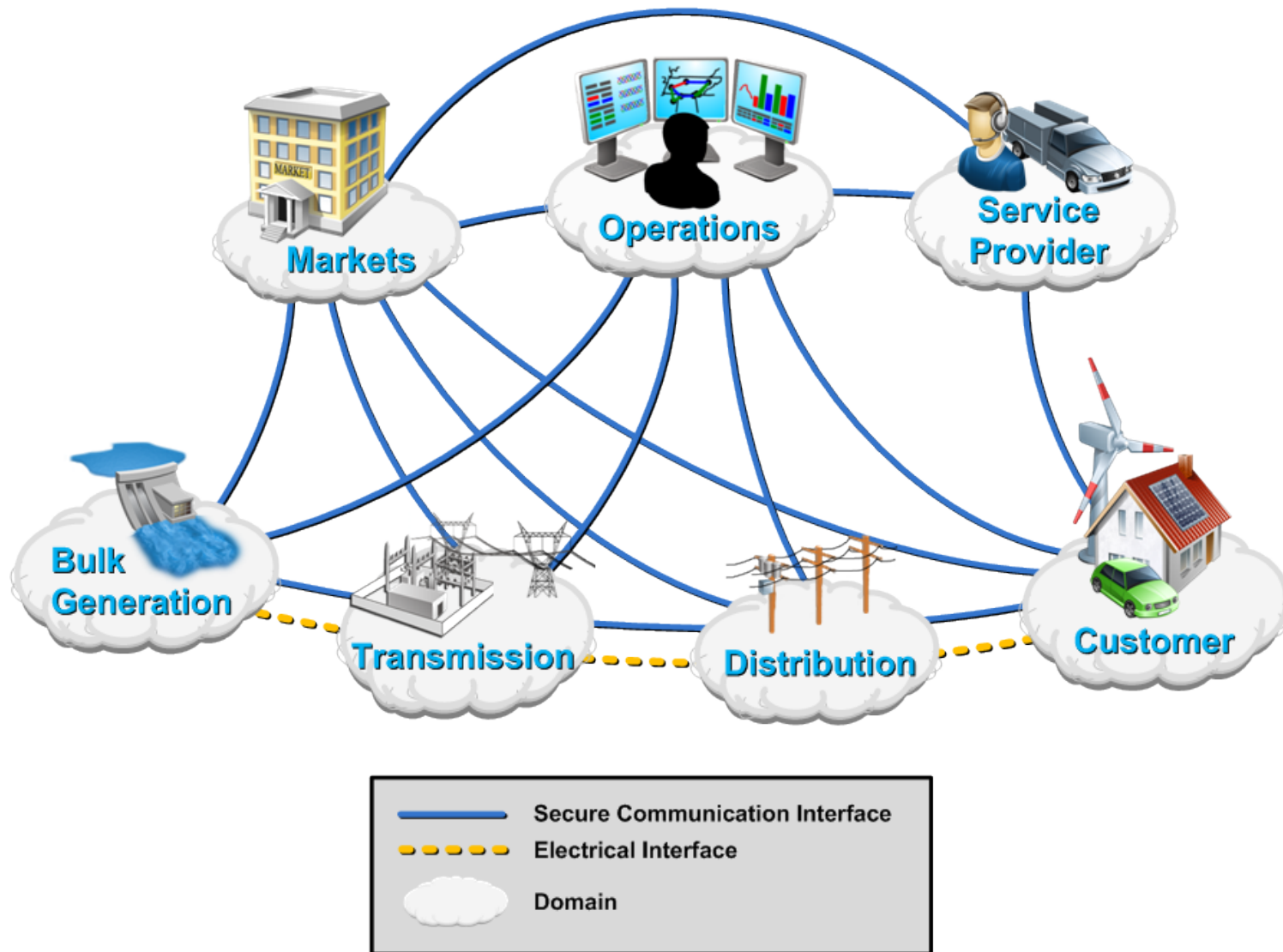
Justin Searle – InGuardians

Types of Security Assessments

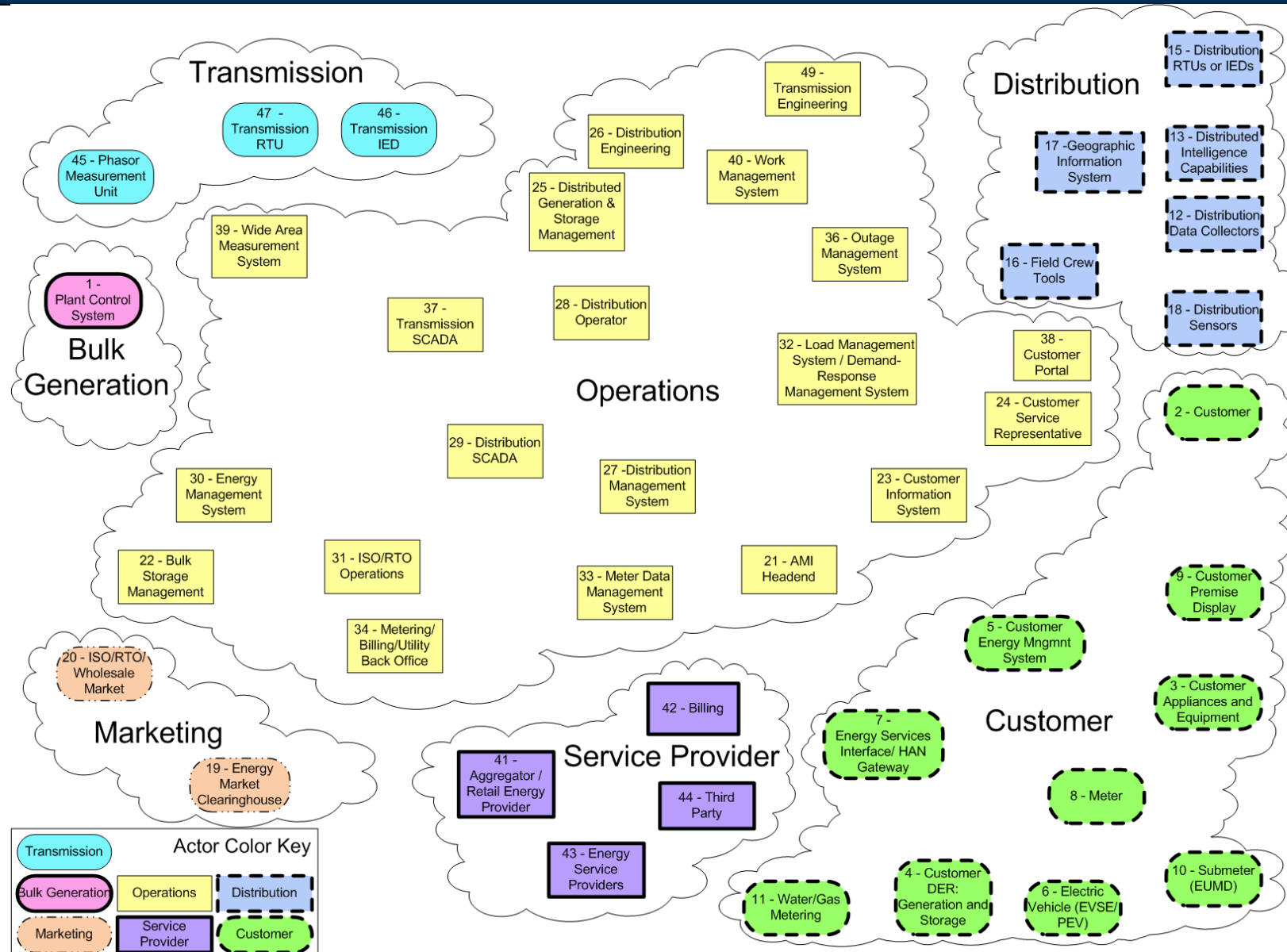


- Best Practice Assessments / Security Architecture Reviews:
 - Focuses on engineer/admin interviews to evaluate security posture
 - Looks at system implementation and configuration
- Vulnerability Assessment:
 - Focuses on the use of automated tools, often with some degree of manual verification
 - Looks for known system vulnerabilities and mis-configurations through the use of vulnerability signatures and system versioning
- Penetration Testing:
 - Focuses on the arts of system misuse and reverse engineering
 - Utilizes the concept of attack and pivot to identify difficult-to-discover vulnerabilities several layers deep
 - More accurately gauges the risk of known vulnerabilities
 - Requires a higher degree of technical expertise and knowledge
 - Extensive manual efforts
 - Custom tool creation

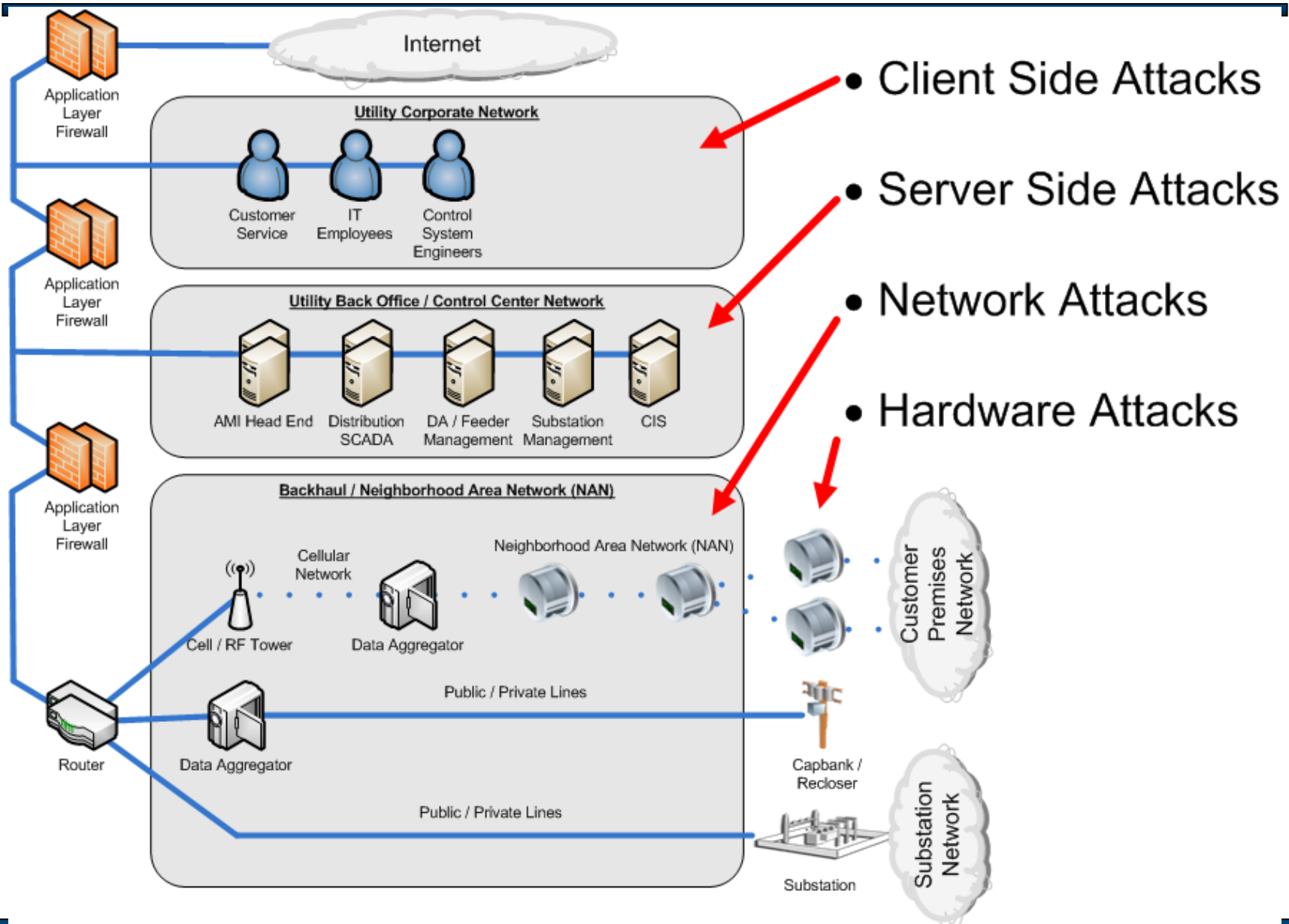
Smart Grid Conceptual Model



Smart Grid Reference Model - Domains



Basic Utility Attack Surface



Client Side Attacks



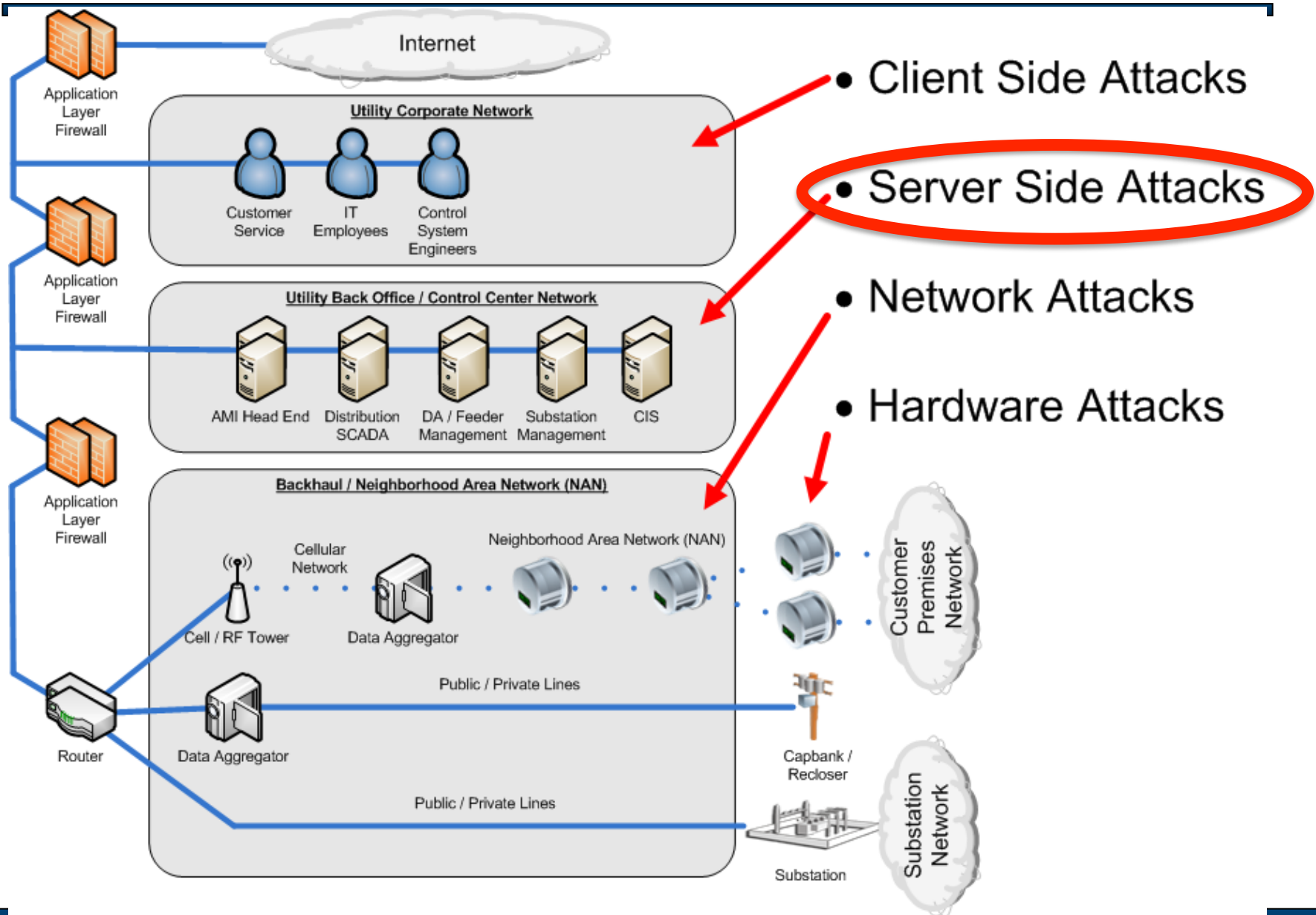
- For years, attackers have been leveraging company workstations as a primary attack avenue
 - Perimeters are getting harder to attack directly
 - Employees are more dependent on the Internet
 - Web browsers have excessive functionality that can be used for both good and evil
 - Employees have access to company's internal systems
- Types of client side attacks:
 - Malware, Viruses, and Botnets
 - Software vulnerabilities via buffer overflows, security boundaries, and software update mechanisms
 - Web browser attacks such as XSS (Cross Side Scripting) to execute malicious code on a user's browser

Client Side Defenses



- Traditional defenses are of limited use against targeted attacks
 - Antivirus can be bypassed within minutes through binary repacking and modification
 - Bypass web proxy filters by using non-blacklisted sites
- Network segregation and properly implemented access control provide the strongest and most economical defense
 - Limit access to sensitive data and control system functionality
 - Segregate sensitive workstations and servers from other systems
- What does this mean for Utilities?
 - Prevent customer service reps from issuing disconnect/reconnect and demand response signals. Have it go through a ticketing system to a smaller control systems team
 - Deny Internet access to all workstations that issue control signals or interface with control systems, such as control center workstations, AMI administrators, and employees approving disconnect/reconnect and demand response signals

Server Side Attacks

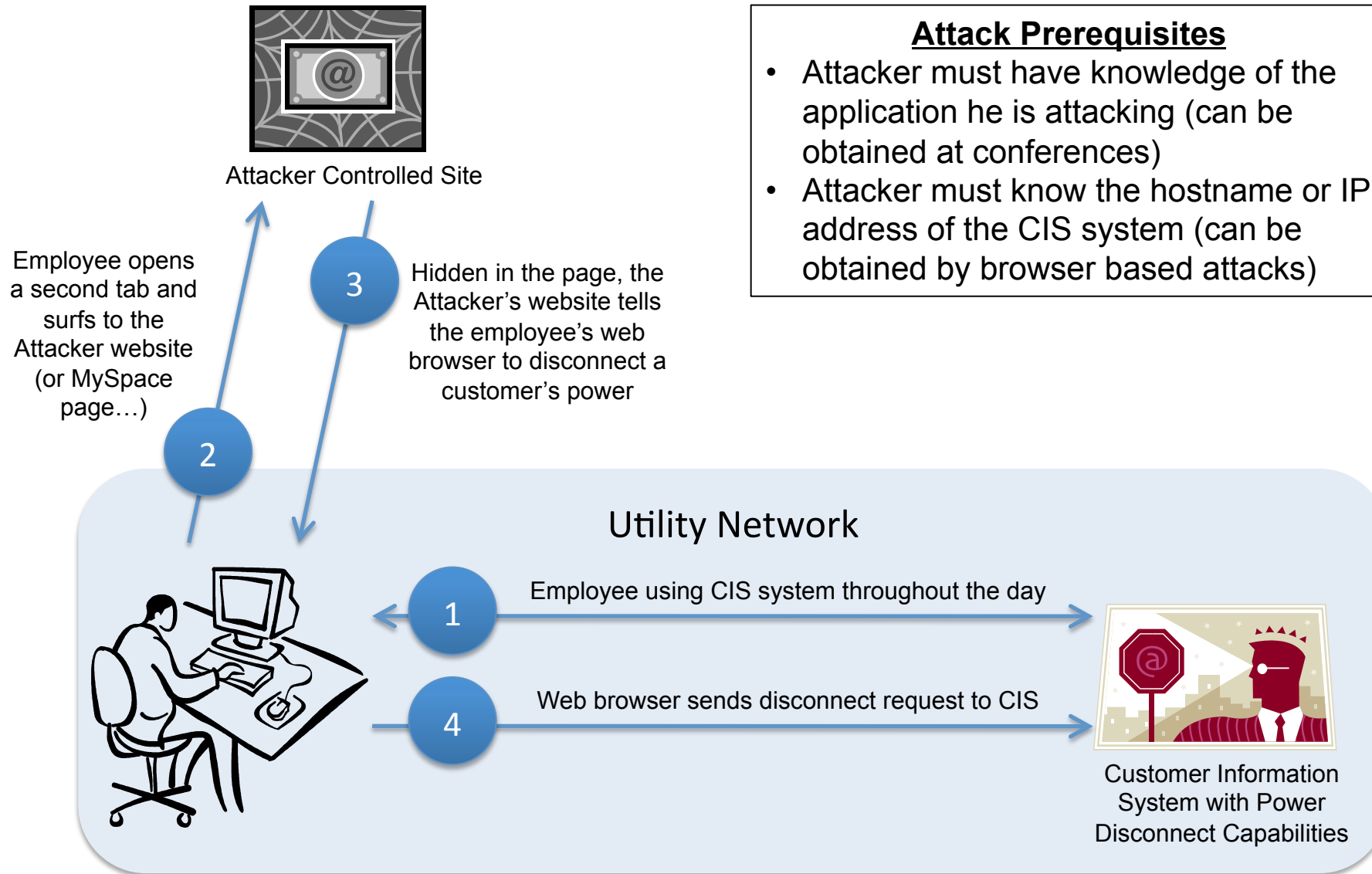


Server Side Attacks



- Customer and Employee portals are obvious targets
- Attacks on internal servers from compromised workstations should also be expected
- Pivoting through internal user web browsers to attack internal web applications is far less obvious
 - Most web applications are vulnerable to CSRF (Cross-Site Request Forgery) attacks
 - CSRF attacks are completely transparent to the user and can affect any system they are currently logged into
 - CSRF attacks don't require compromised workstations
- It is critical to understand web based attacks like CSRF because most of our Smart Grid systems use web based management interfaces

Cross-Site Request Forgery (CSRF)

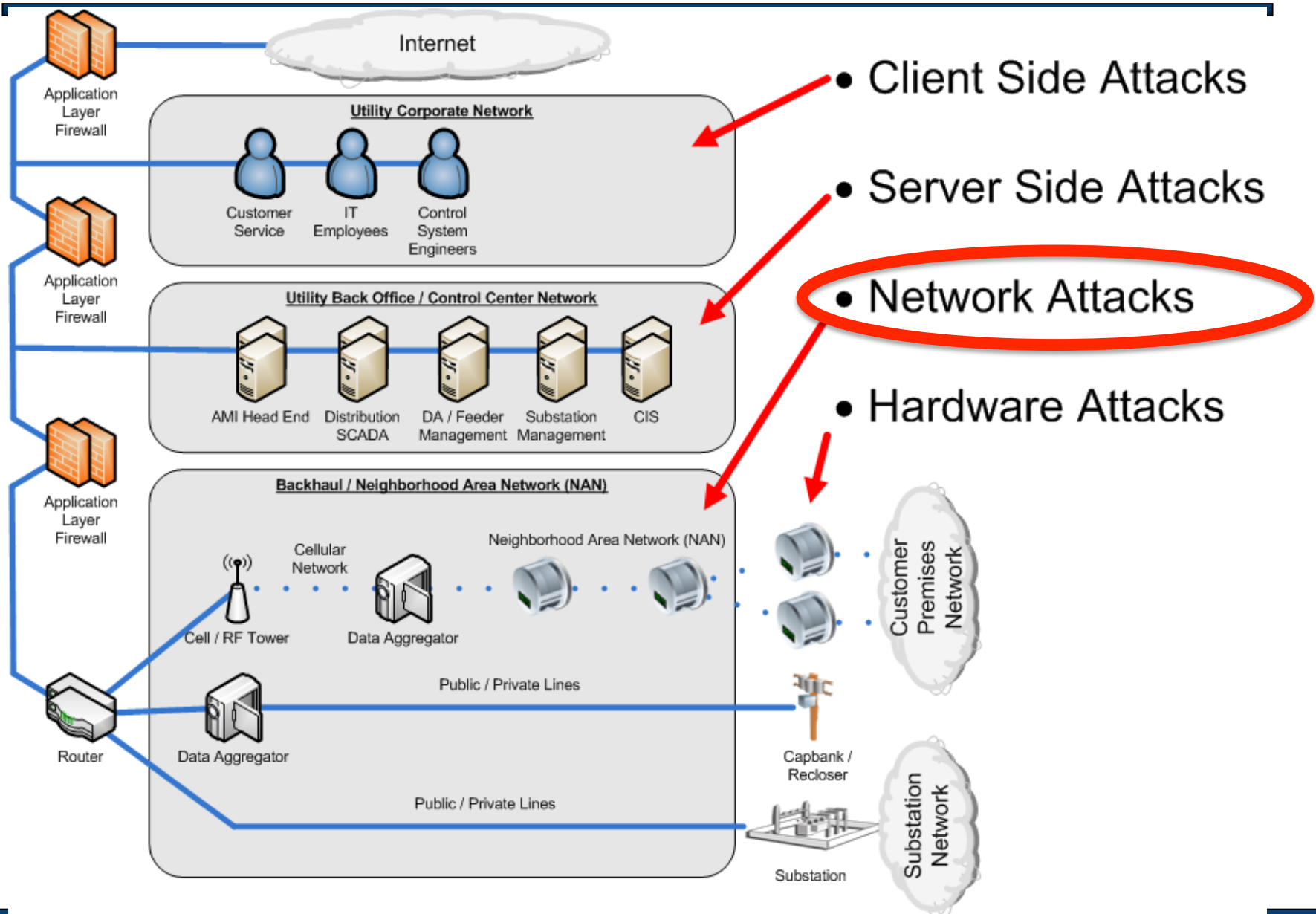


Server Side Defenses



- Keep systems patched and updated
- Perform periodic vulnerability assessments and penetration tests
- Use Intrusion detection and intrusion prevention systems in strategic positions around highly sensitive servers and control management systems
- Utilize centralized logging systems for alerting and forensic evidence

Network Attacks



Network Protocols and Security



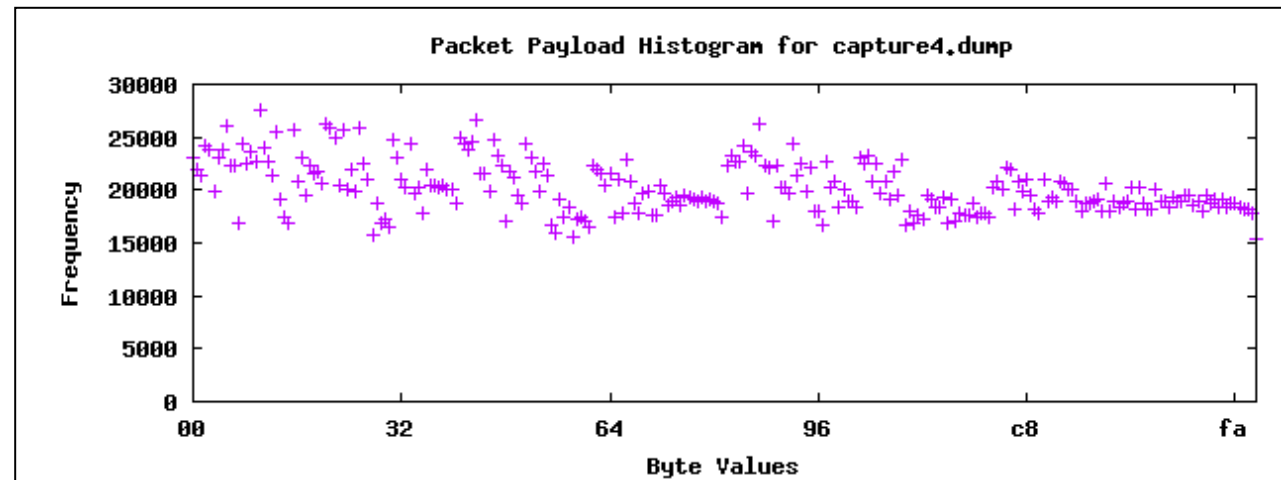
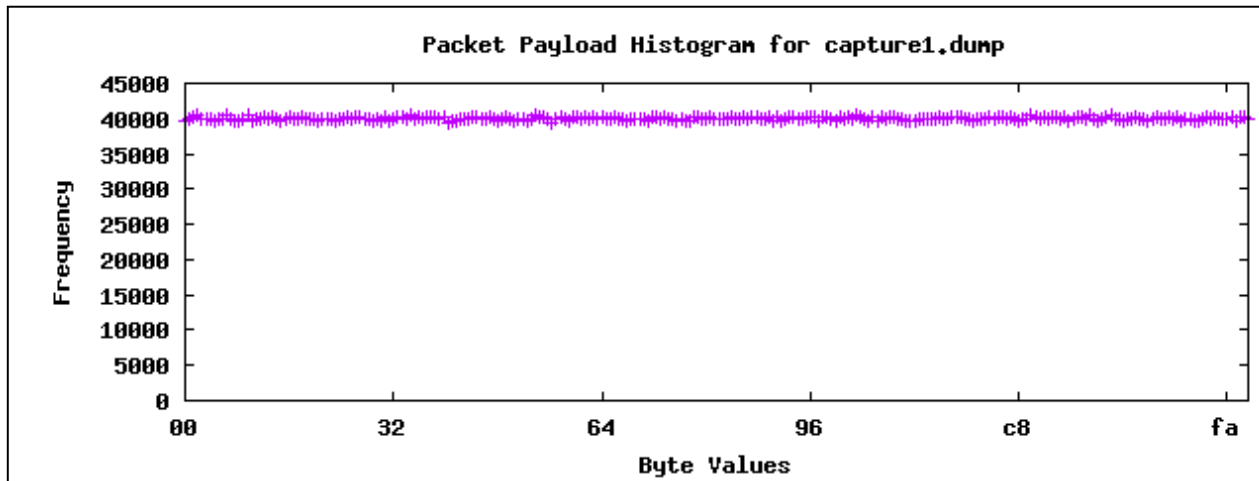
- It's pointless to compare proprietary protocols to standards based protocols from a security perspective
 - standards based protocols benefit from greater transparency, but suffer from “interoperable” hacker tools
 - proprietary based protocols benefit from obscurity and sparse hacker tools, but suffer from limited security reviews
 - the same arguments can be made for open source vs. proprietary software
- Securely architected protocols is essential, but properly implemented and configured protocols are just as important

Attack: Weak Cryptography



- Many proprietary systems implement their own cryptography
 - Some create their own crypto algorithms
 - Others create their own crypto stacks of known algorithms
 - Just because it's "AES" doesn't mean it's secure
- Exploits in insecure cipher modes, weak nonce construction, IV re-use, etc...
- Practical attacks include replaying data, decrypting packets, key recovery, data manipulation / injection
- Analysis tools to test implementations: Ent, visualization of RNG's, cryptographic accelerators, custom scripts

Histogram 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
```

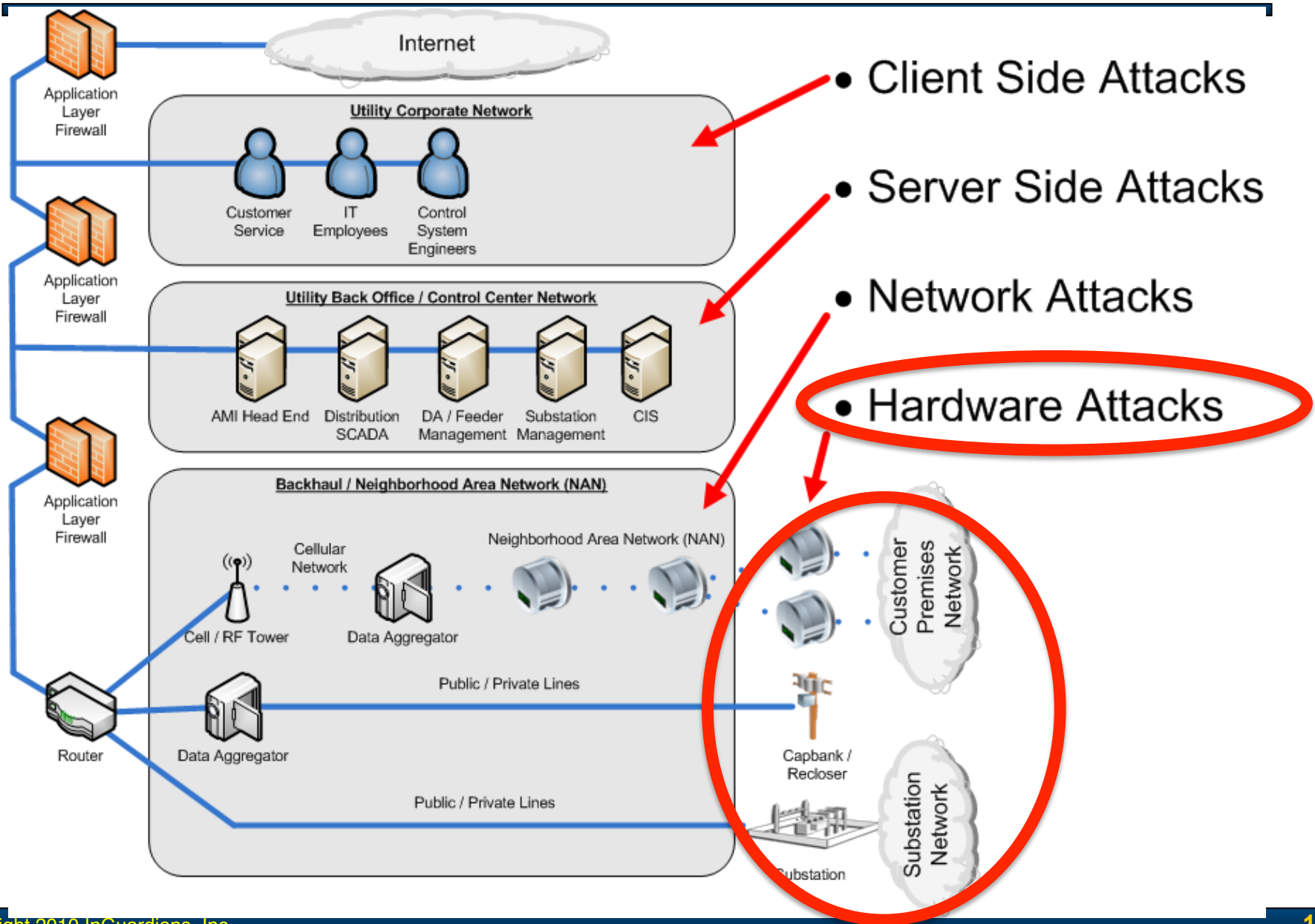
Defense: Weak Cryptography



- Design and implementation of cryptographic systems is extremely difficult
 - Avoid this if possible
 - Leverage vetted third-party encryption stack implementations
- If necessary, model system after proven protocols
 - IEEE 802.11i RSN key derivation
- Expert cryptographic review consulting

Vulnerabilities in crypto are especially hard to recover from
(remember WEP?)

Hardware Attacks



Hardware Attacks



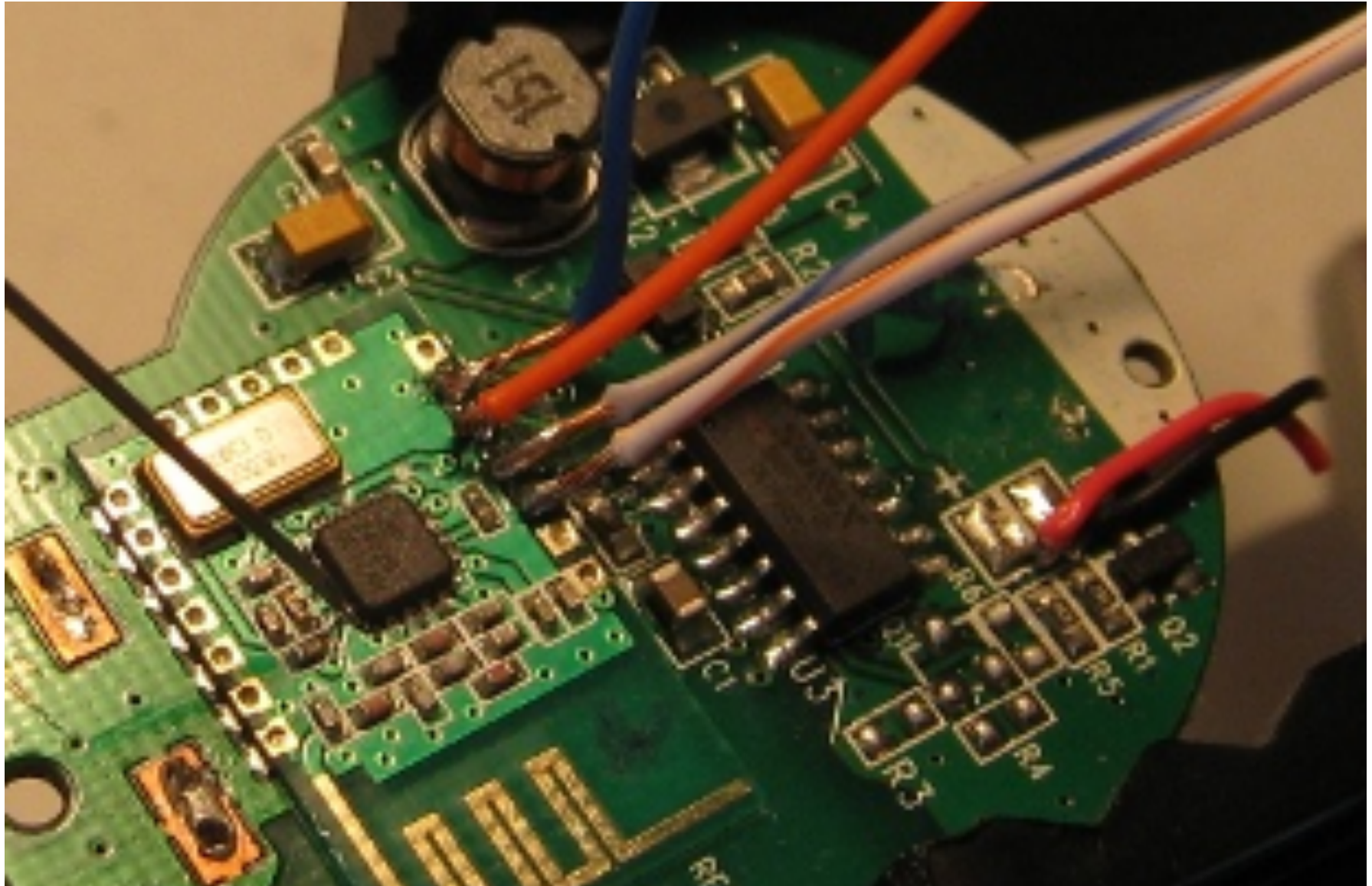
- All field deployed devices are susceptible to physical hardware attack
 - Meters on residential homes are obvious targets
 - Pole-top devices such as DA and feeder automation devices are not much harder to access (albeit riskier to health)
 - Substation physical defenses are a deterrent, not an insurmountable obstacle
- If tamper mechanisms or perimeter alarms are triggered, modified hardware is not easily detected
- Basic Hardware Attacks:
 - Encryption key and flash extraction
 - Firmware / Software vulnerabilities
 - Flash image manipulation

Attack: Key & Firmware Extraction

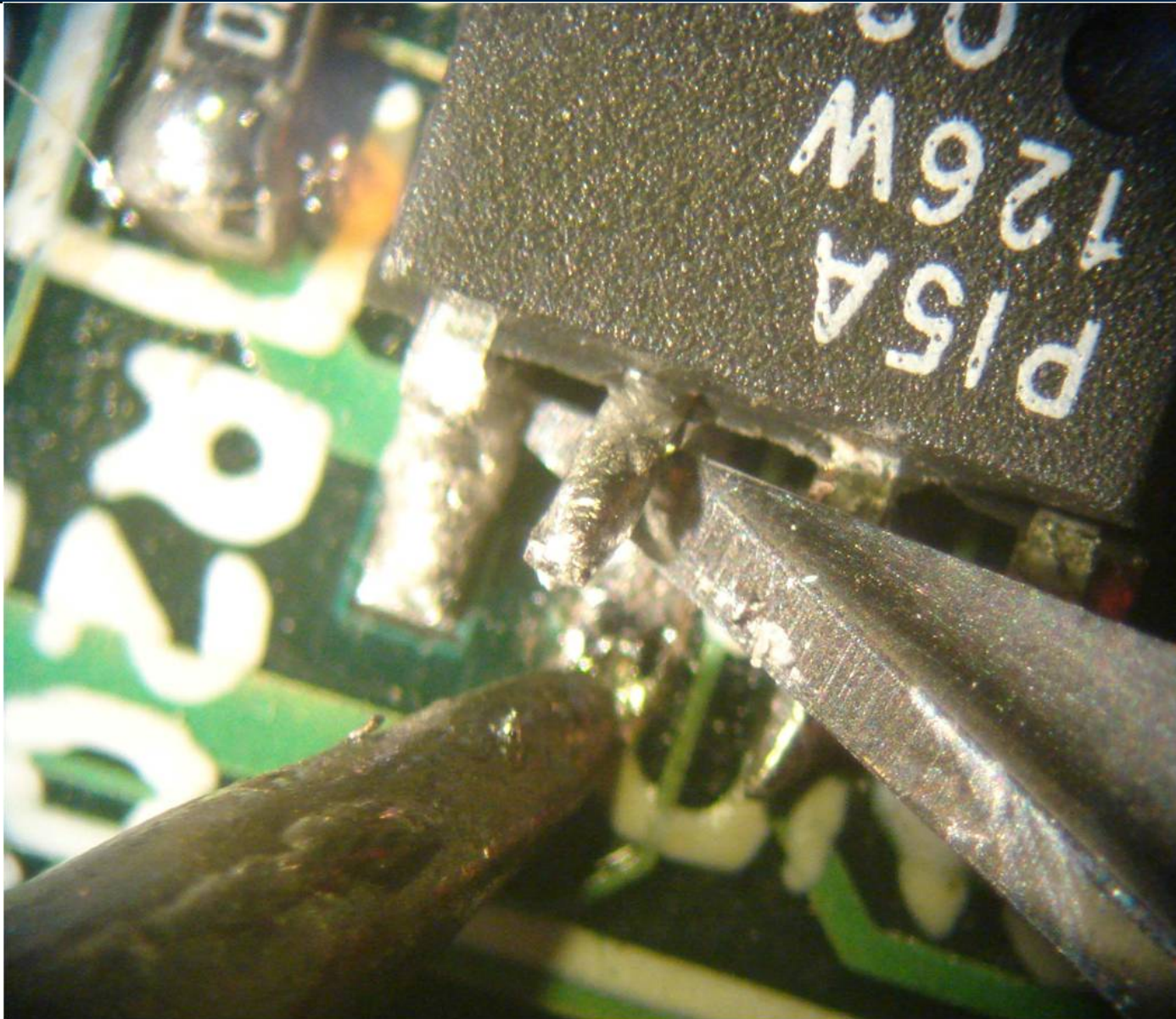


- Extract locally stored encryption key and firmware
 - Extract contents of RAM, Flash, and EEPROM data
 - Identify encryption key or firmware
 - Especially useful when a single key is shared across multiple devices
- Intercepting data between circuit board peripherals
 - Operate and boot device normally in a lab, monitoring bus activity between major chips (MCU, Radio, EEPROM, RAM)
 - Identify encryption key or firmware
 - Encryption key can often be found in key load operations between a microcontroller and crypto accelerator
 - Firmware can often be found in software updates between radio and flash

Interfacing with an IC



Lifting an IC's Chip Enable (CE) Pin





SPI Bus Snooping



spi-eeprom - Total Phase Data Center

22.77 KB

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 stopped [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 stopped [Wed May 13 16:19:39 2009]	

Search

No filter: 64 records.

Protocol Lens: SPI

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
0000	02	00	00	00	01	02	03	04	
0008	05	06	07	08	09	0A	0B	0C	
0010	0D	0E	0F	10	11	12	13	14	
0018	15	16	17	18	19	1A	1B	1C	
0020	1D	1E	1F						...
0028									
0030									
0038									

MOSI Data MISO Data Timing

Bus Filter Info

Disconnected EN

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

Symmetric Key Search



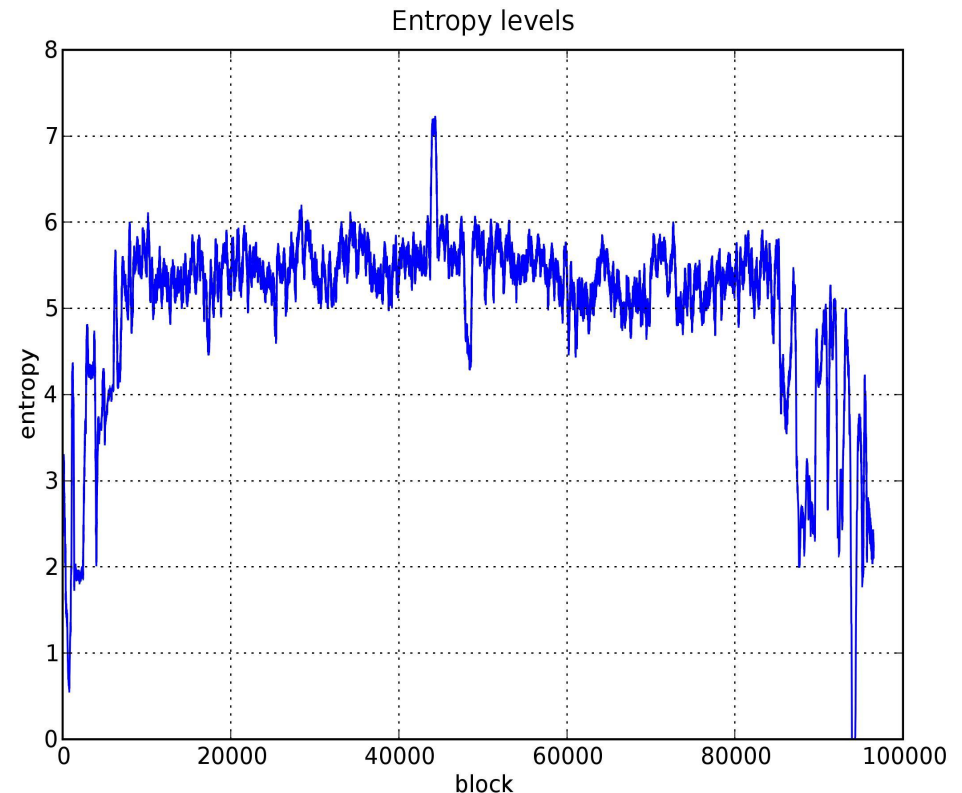
- Basic string searches for obvious keys
- Develop custom tools to do more advanced searches:
 - GoodFET: Abuses vulnerability in TI, Ember radios to access RAM even when chip is locked
 - zbgoodfind: Search for ZigBee key using RAM dump as a list of potential keys
 - Combined they can recover the ZigBee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

Asymmetric Key Search



- 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



Defense: Key & Firmware Extraction



- Utilize System-on-a-Chip (SoC) devices when possible
- Hardware tamper-proof mechanism and monitoring
 - Learn from Microsoft, epoxy layers are only a speed bump
- Limit encryption key distribution to small groups of devices, preferably with unique keys per meter
- Obscure encryption key storage
- TPM's can protect asymmetric keys
- Implement key rotation mechanisms

Be prepared to answer: What is my remediation strategy once the encryption keys protecting the NAN are compromised?

Conclusion



- Required skills for assessing Smart Grid security cover many areas
 - Hardware, software, wireless, cryptography and more
- Through efficient testing, we can address vulnerabilities before they threaten deployments
- Publically available AMI Attack Methodology
 - Download it at www.inguardians.com
 - An InGuardians created document funded by the original ASAP (AMI Security Acceleration Project) project
 - Provides a detailed methodology for performing penetration tests on smart meter networks
 - Methodology can be adapted for Feeder automation and Substation networks

Contact Information



Justin Searle
justin@inguardians.com
801-784-2052
www.inguardians.com