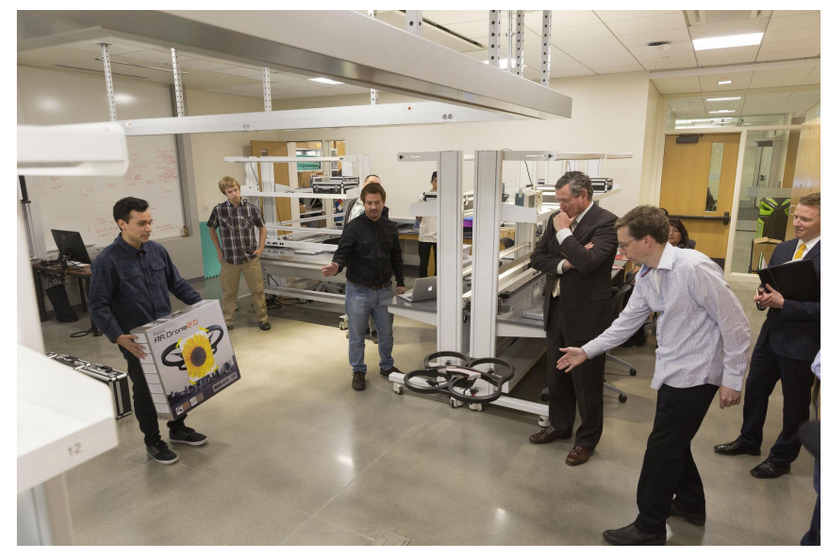
Cybersecurity at CI

Michael Soltys
Professor and Chair of Computer Science

October 18, 2018
Point Mugu

NAWCWD
Cybersecurity Collaboration

California State University at Channel Islands: Department of Computer Science



Outline

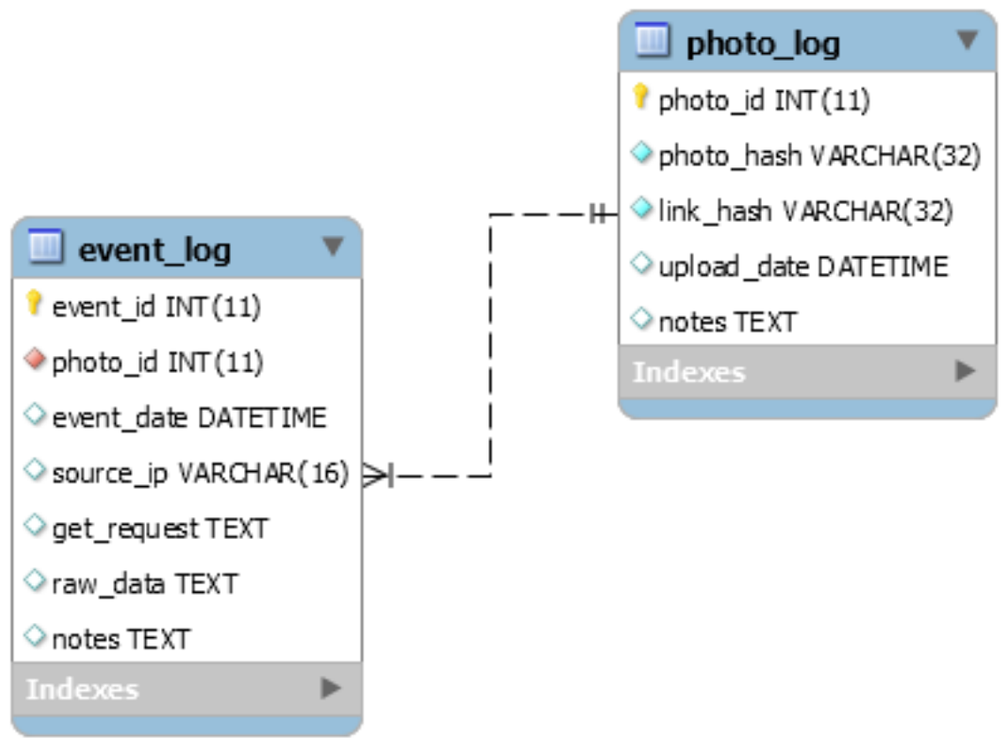
Cybersecurity at CI

Digital Forensics
Collaboration with HTTF

Penetration Testing

SEAKER

Voyager


photo_log
photo_id INT(11)
photo_hash VARCHAR(32)
link_hash VARCHAR(32)
upload_date DATETIME
notes TEXT
Indexes
event_log
event_id INT(11)
photo_id INT(11)
event_date DATETIME
source_ip VARCHAR(16)
get_request TEXT
raw_data TEXT
notes TEXT
Indexes

Password Beast

Virus Total

VirusTotal

AWS

COMP 424/524

3 Klein's attack on RC4

Suppose w key streams were generated by RC4 using packet keys with a fixed root key and different initialization vectors. Denote by $\mathsf{K}_u = (\mathsf{K}_u[0], \ldots, \mathsf{K}_u[m]) = (\mathsf{IV}_u \| \mathsf{Rk})$ the uth packet key and by $\mathsf{X}_u = (\mathsf{X}_u[0], \ldots, \mathsf{X}_u[m-1])$ the first m bytes of the uth key stream, where $1 \leq u \leq w$. Assume that an attacker knows the pairs $(\mathsf{IV}_u, \mathsf{X}_u)$ – we shall refer to them as *samples* – and tries to find Rk.

In [5], Klein showed that there is a map $\mathcal{F}_i \colon (\mathbb{Z}/n\mathbb{Z})^i \to \mathbb{Z}/n\mathbb{Z}$ with $1 \leq i \leq m$ such that

$$\mathcal{F}_i(K[0], \ldots, \mathsf{K}[i-1], \mathsf{X}[i-1]) = \begin{cases} \mathsf{K}[i], & \text{with Prob} \approx \frac{1.36}{n} \\ a \neq \mathsf{K}[i], & \text{with Prob} < \frac{1}{n} \text{ for all } a \end{cases}$$

If the first i bytes of a packet key are known, then the internal permutation S_{i-1} and the index j at the $(i-1)$th step of the RC4 key setup algorithm can be found. We have

$$\mathcal{F}_i(\mathsf{K}[0], \ldots, \mathsf{K}[i-1], \mathsf{X}[i-1]) = \mathsf{S}_{i-1}^{-1}[i - \mathsf{X}[i-1]] - (j_{i-1} + \mathsf{S}_{i-1}[i]) \bmod n$$

The attack is based on the following properties of permutations.

Theorem 1 *For a random permutation* P*, and random number* $j \in \{0, \ldots, n-1\}$*, we have*

$$\text{Prob}(\mathsf{P}[j] + \mathsf{P}[\mathsf{P}[i] + \mathsf{P}[j] \bmod n] = i \bmod n) = \frac{2}{n}$$

$$\text{Prob}(\mathsf{P}[j] + \mathsf{P}[\mathsf{P}[i] + \mathsf{P}[j] \bmod n] = c \bmod n) = \frac{n-2}{n(n-1)}$$

where $i, c \in \{0, \ldots, n-1\}$ *are fixed, and* $c \neq i$.

AWUS 036 ACH

Course Outline

1. Crypto: basics
2. Crypto: symmetric ciphers, Assignment: break a MAC
3. Crypto: DES, IDEA, AES, htpasswd; case study: break WEP
4. Crypto: blocks & hashes
5. Crypto: public key: D-H, ElGamal, RSA, elliptic curves
6. Authentication: kerberos
7. Tools:

   I. OpenSSL & GnuPG

   II. Hashcat & John Ripper

   III. Kali Linux, Wireshark & Palo Alto Firewalls

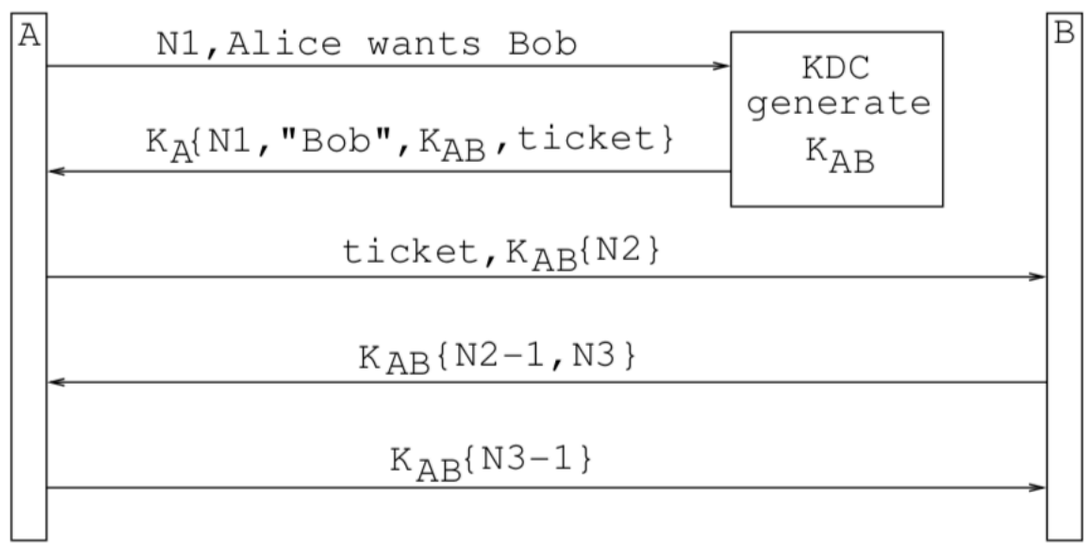
   IV. Malware

Cryptoanalysis
How to break a code
Encoding vs Encryption

Base 64 Encrypted String:

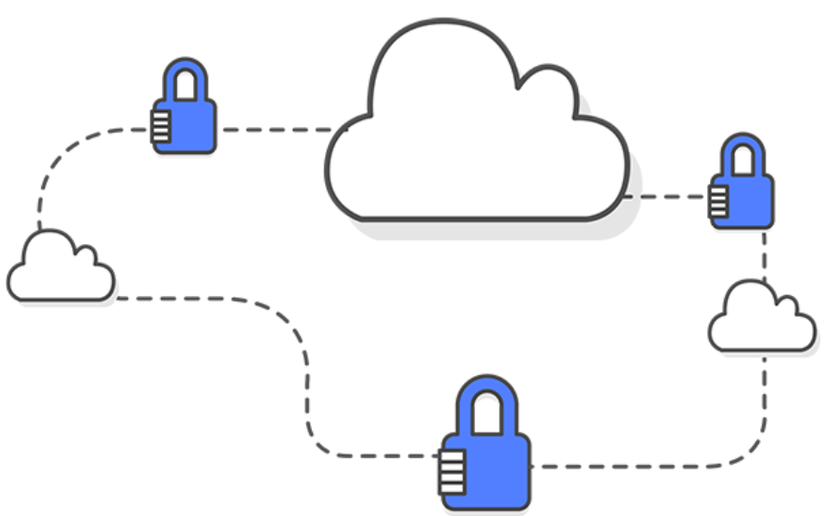

```
oZ04bYkMsJoFV3csbYwcmXEcjnHzj3EUlHndV4kMsJoFbYkMsJ
oFeVU1wqcBtpM6bZcFtpw7wVDzV34Bbak7slU5uKc4vKkGbZQ5
bak7slUBtpw7wW/zV4w7rqjztpIAuKcHrpDztZYBsVUCv1U4xp
n/bT8WuKo/sVU5v5YAslUHta3zs5o0v5sIuVUGxpIAsqkFxmPz
Vz8cu1UKtZYHbZk8vKk0u6jzsZo4vafzuKbzvJA8sqfBbT8Vwq
cBwVUHtZnzs54FslUCs1UHtZ4BslU4xpoGiFTdmJLzwJ00wVUK
tpM6vFU3rqc4bZ04bZYGvZ4FsmPzV4w7rqjzwZ04bZ00u5j/bZ
k0v5nzvJo8x5nzwZ04bZs8v5nSbT7djpM3baw7rqjzvJ0CwpE3
sqb/bVrzwJ00wVU0v6j/bT8WuKo/sVUHwJ4GwVUHtZnzvJ4Bsq
wGbZQ5bak7xlU7spYFwWPzV3YBsVUKtZoBbak7xlU7spYFwVU1
spw0u1UHuFU1spYHeVTdoJ00wVU3v5o0sVU7rpM3iFT5baw7rq
jzsac4rpjzs5o4wWPzVz8qtZYHbak7slU7rpIAsqbSbaw7rqjz
wZ04bZg7rp4BeVTdlpLzwJ00wVU5wqcBrpg4baw0vFUHta3zr6
c0tpLSbT8qtZYHbak7slU0u6s8uWPzwJ00wVU3v5o0sVU6v5YG
vVDzV3k0v5nztqkGbZk4rpk/xlUHsqcFuKcGbZg/rqgDblTdV4
w7spLzwZ04bagHrqcGbak7v5oKbZkCwJLzwZ04tqbzvKU4rqcG
bT8Uu5jzwJYHsqb6sVU7spYJspLzwJ4HtVUHtZo8v1UHspYFvG
7zV3k8sVU7slUGup4/slU7tqfzwJQFtFUHuFUGspnSbT8Xtpjz
tZnzwJ0CbZI0sZnzwZ04bXE0upbzupY+slUHtZo4iFTdV4kMsJ
oFbYkMsJoFbZcIv5M8u5vzr6c8sJ0HeVTdlpLzwZ04bZsCv5oG
wafzuJrzwZ04bZM8sJ0Hh1TdoJ00wVU8upICv6k0uVU7rpM3bZ
QFbZoMslDzV3k0v5nzs6c0upnzwZ0MbZs4rqc5wpDzvK4AupoH
v63SVz==
```

Needham-Schroeder Protocol



8. CompTIA ASecurity +

9. AWS security



10. Another assignment, usually coding in Python

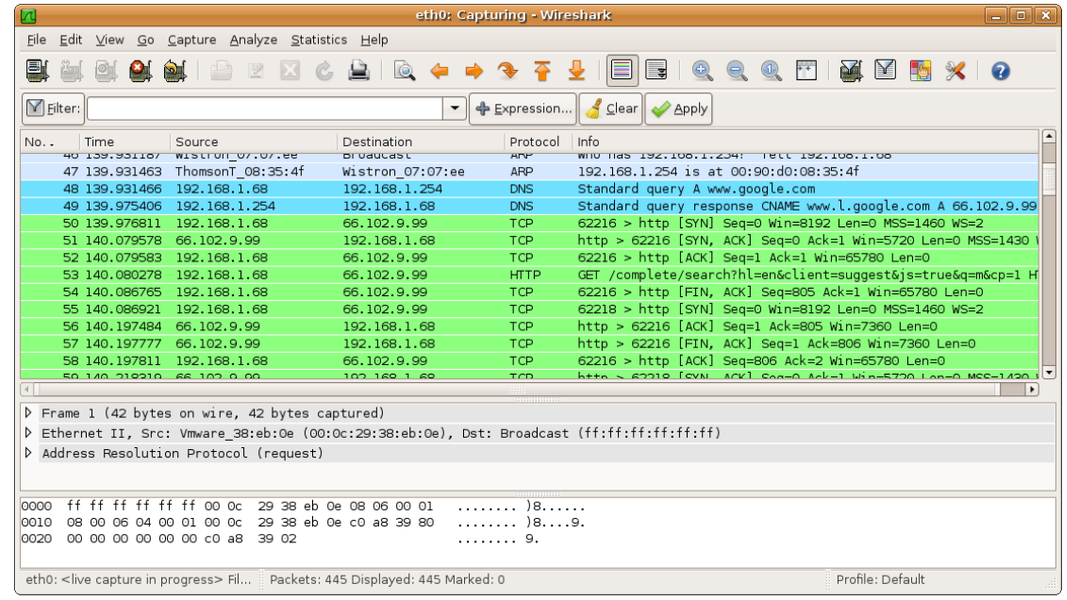
11. Presentations

    I. Long list of articles (articles more current than textbooks)

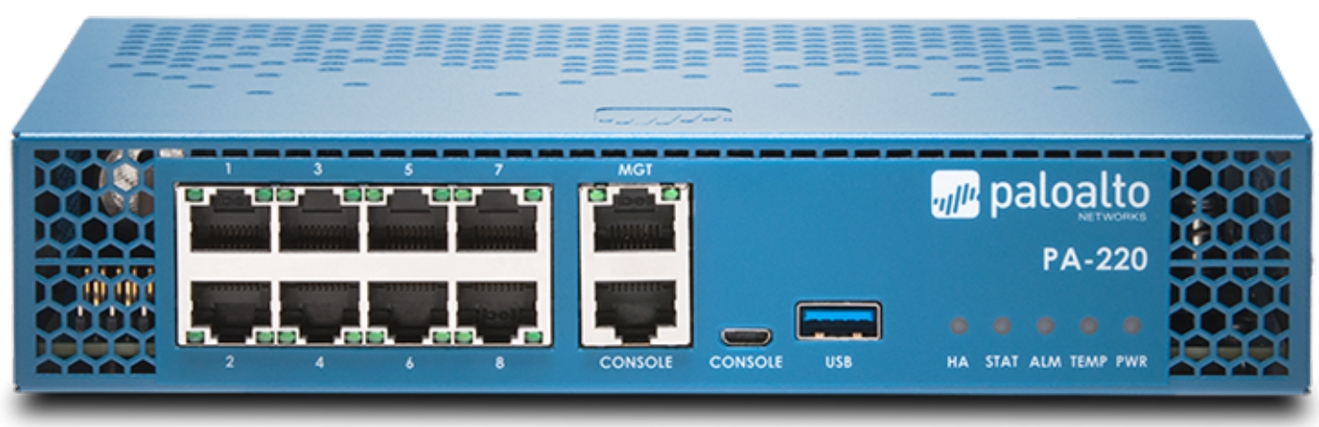
    II. Ability to articulate ideas about security

Penetration Testing

Kali Linux



Wireshark



Palo Alto Firewall

Penetration testing tools on Kali Linux

1. Nmap
2. Aircrack-ng
3. Wifiphisher
4. Burp Suite
5. OWASP ZAP
6. SQLmap
7. CrackMapExec
8. Impacket
9. PowerSploit
10. Luckystrike
11. BeEF
12. THC-Hydra
13. Immunity
14. Social Engineering Toolkit

Questions?

Michael Soltys

michael.soltys@csuci.ed
www.msoltys.com
@MichaelMSoltys