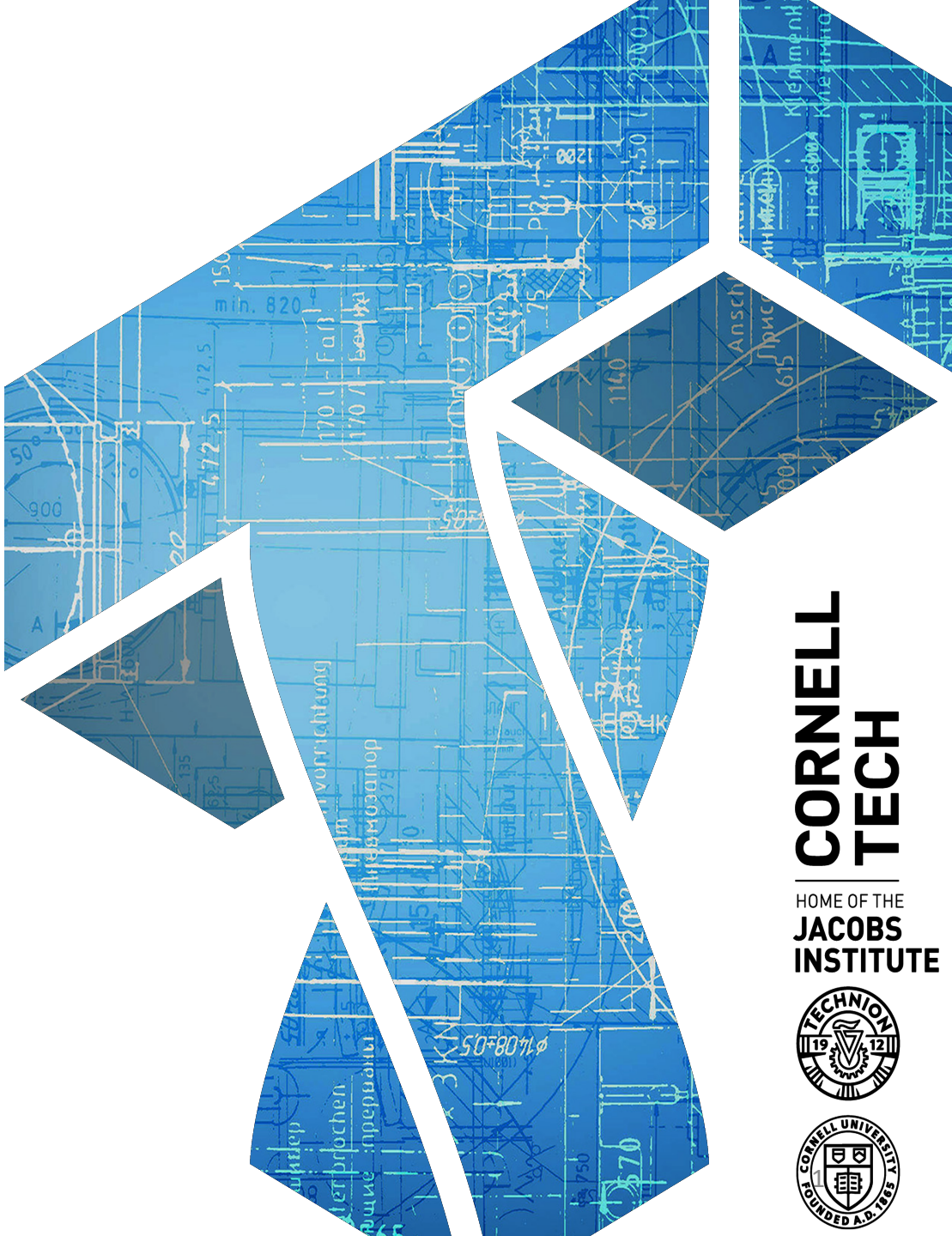
CS 6431: Cryptographic backdoors

Instructor: Tom Ristenpart

https://github.com/tomrist/cs6431-fall2021



CORNELL TECH
HOME OF THE JACOBS INSTITUTE

Nation-state surveillance

- Governmental intelligence agencies spy on each other
- End-point compromise, hacking operations
  - Government agency examples:
    - Tailored Access Operations (TAO) & FOXACID (USA)
    - Unit 8200 (Israel)
    - PLA Units 61398, 61486 (China)
    - Fancy Bear / GRU (Russia)
  - Commercial examples:
    - FinSpy by FinFisher Gmbh (Germany)
    - NSO Group's Pegasus (Israel)
    - DarkMatter (UAE)

Targeted attacks

- Dissidents, journalists, activists targeted by nation-states
  - Phishing attacks, botnet-style C&C servers to collect data
  - Remote Access Trojans (RATs)
- Small industry of companies providing “lawful access” tools


From: Melissa Chan <melissa.aljazeera@gmail.com>
To:
Sent: Tuesday, 8 May 2012, 8:52
Subject: Torture reports on Nabeel Rajab

Acting president Zainab Al Khawaja for Human Rights Bahrain reports of torture on Mr. Nabeel Rajab after his recent arrest.

Please check the attached detailed report along with torture images.

1 attachment: Rajab.rar 1.4 MB Save

Figure 1: E-mail containing FinSpy.

https://www.usenix.org/system/files/conference/usenixsecurity14/sec14-paper-blond.pdf
https://www.usenix.org/system/files/conference/usenixsecurity14/sec14-paper-marczak.pdf
https://www.usenix.org/system/files/conference/usenixsecurity14/sec14-paper-hardy.pdf

Tracking GhostNet (2009)

http://www.nartv.org/mirror/ghostnet.pdf

- In-depth investigation into espionage campaign against Tibetan community
- Discovered malware operating on Tibetan assets
- Reverse engineered, found C&C servers
  - No authentication to access C&C servers
  - Lists all computers infected by malware
- 1,295 infected computers in 103 countries
  - 26.7% government- or politics-related

2009 Operation Aurora

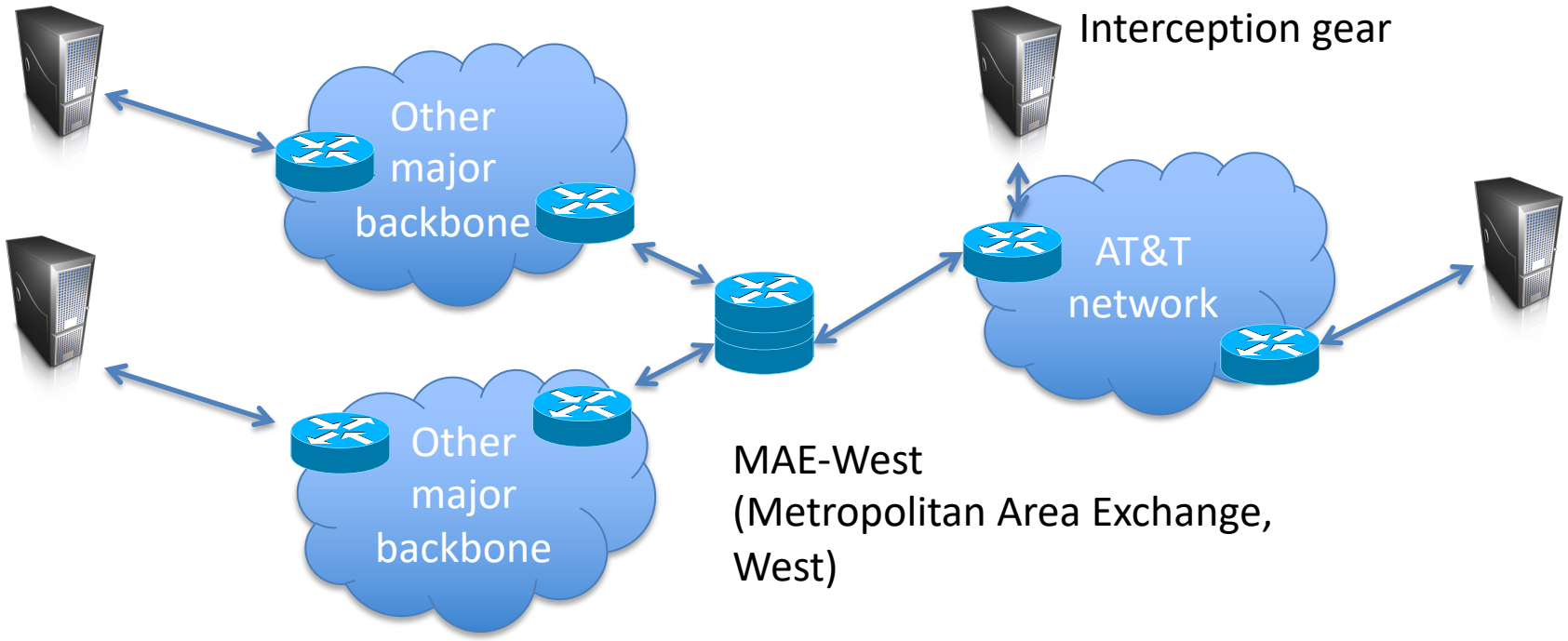
- Elderwood advanced persistent threat group (alleged ties to PLA)
  - "Waterhole attacks" (compromise site target company's employees visit)
- Targeted systems at Google (IP theft, Gmail accounts of Chinese dissidents), Adobe, Juniper, Rackspace, Microsoft, …
- One theory: alleged counterespionage campaign
  - Figure out what Gmail accounts being monitored by US Government
  - Remember: CALEA/FISA lawful intercept

NSO Pegasus

- Ahmed Mansoor (UAE) human rights activist
  - "On August 10 and 11, 2016, Mansoor received SMS text messages on his iPhone promising "new secrets" about detainees tortured in UAE jails if he clicked on an included link."
- Analysis indicated that link connects to exploit chain to remotely jailbreak iPhone
  - Three zero-days (WebKit browser vuln, ASLR bypass, kernel vuln)
  - Remotely installs implant (spyware)

https://citizenlab.ca/2016/08/million-dollar-dissident-iphone-zero-day-nso-group-uae/

Wiretap surveillance



Large amounts of Internet traffic cross relatively few key points

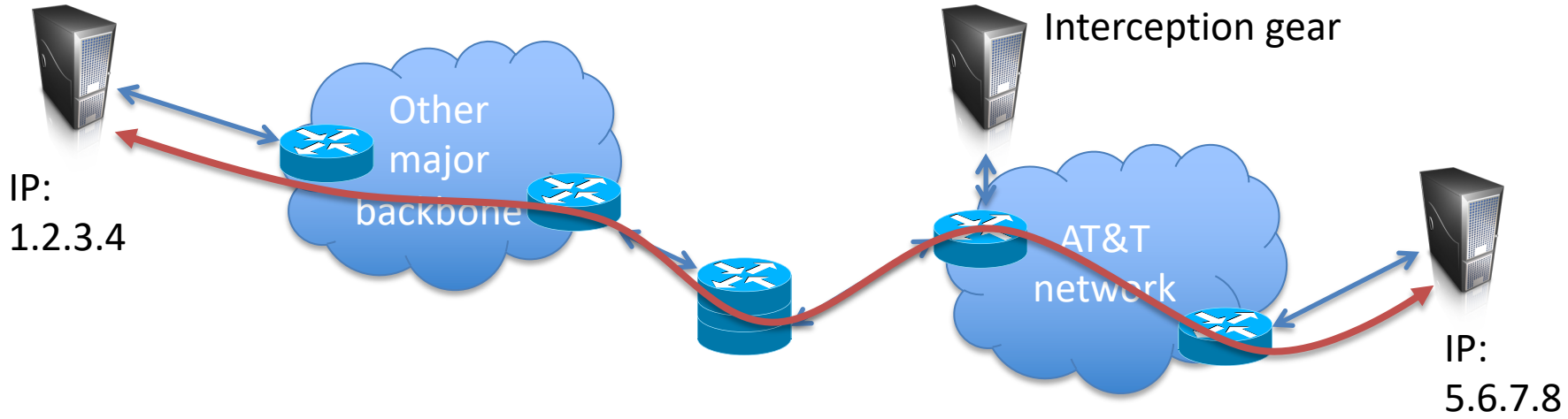
AT&T Wiretap case

- Mark Klein discloses potential wiretapping activities by NSA at San Francisco AT&T office
- Fiber optic splitter on major trunk line for Internet communications
  - Electronic voice and data communications copied to “secret room”
  - Narus STA 6400 device

Lawful intercept

- CALEA
  - Communications Assistance for Law Enforcement Act (1995)
- FISA
  - Foreign Intelligence Surveillance Act (1978)
  - Demark boundaries of domestic vs. foreign intelligence gathering
  - Foreign Intelligence Surveillance Court (FISC) provides warrant oversight (good example of regulatory capture)
  - Executive order by President Bush suspend need for NSA to get warrants from FISC
- Almost all national governments mandate some kind of lawful intercept capabilities

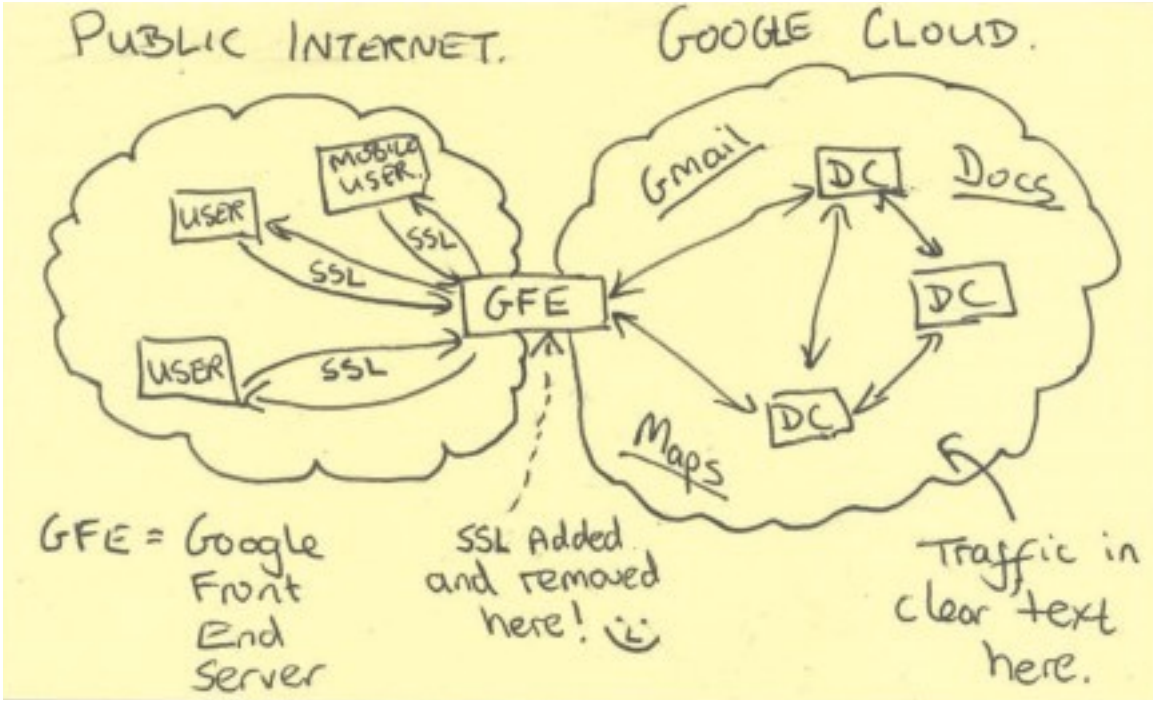
Encryption and intercept



- TLS, SSH, IPSec all prevent interception of plaintext data
- Prevents wiretapping for content
- What to do about it?

Snowden revelations

- HTTPS terminated at edge of Google networks
- Internal data center-to-data center communications on privately leased lines
  - No encryption up until summer 2013



History of overt weakening

- DES key length reduced at NSA's bequest
- Clipper Chip in 1990s
  - NSA-designed encryption chip
  - Secret keys given at factory, escrowed with NSA
  - Campaign by cryptographic experts, others to prevent use
- Export controls (1990s and on)

Covert sabotage of crypto

- Purposefully inserting weaknesses into cryptographic protocols and/or implementations
- United States' NSA has history of doing so
  - Dual EC PRNG case probably most well known

Desiderata for good sabotage:

- Allow decryption, ideally in real time
- Decryption should be private
  - Only saboteur should be able to exploit
- Undetectability
- Others?

See [Schneier et al. 2015] for taxonomy and easy-to-read summary

TLS 1.2 handshake for RSA transport

Client

Server

Pick random Nc

ClientHello, MaxVer, Nc, Ciphers/CompMethods →

Pick random Ns

← ServerHello, Ver, Ns, SessionID, Cipher/CompMethod

Check CERT using CA public verification key

← CERT = (pk of Amazon, signature over it by CA)

Pick random PMS
C <- Enc(pk,PMS)

C →

PMS <- D(sk,C)

ChangeCipherSpec,
{ Finished, PRF(MS, “Client finished” || H(transcript)) } →

← ChangeCipherSpec,
{ Finished, PRF(MS, “Server finished” || H(transcript’)) }

Bracket notation means contents encrypted

MS <- PRF(PMS, “master secret” || Nc || Ns)

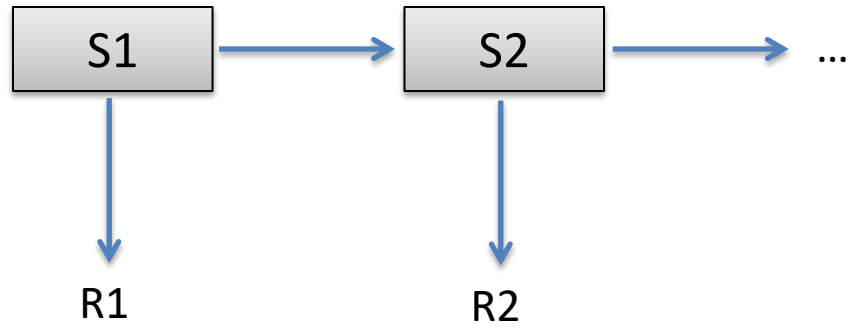
Sabotaging PRNGs

- Say we can sabotage client's random numbers to make them predictable
- Where do random numbers come from?
  - Use system service like /dev/urandom to generate initial seed S1
  - Use S with a pseudorandom number generator (PRNG)

(S2, R1) <- PRG(S1)

(S3, R2) <- PRG(S2)

⋮



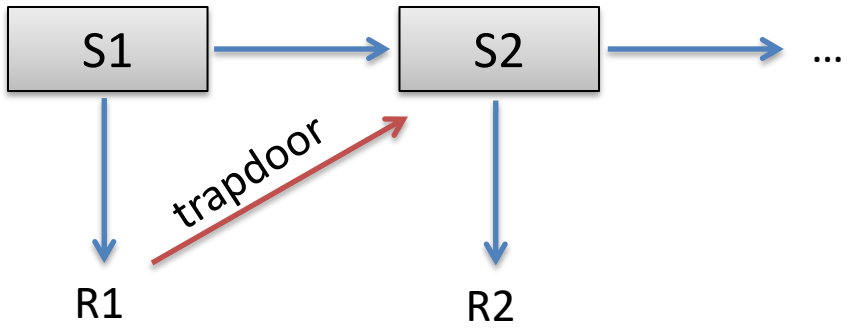
Sabotaging PRNGs

- Arrange that given R1, attacker with a trapdoor can compute S2
- This allows predicting all subsequent values

(S2, R1) <- PRG(S1)

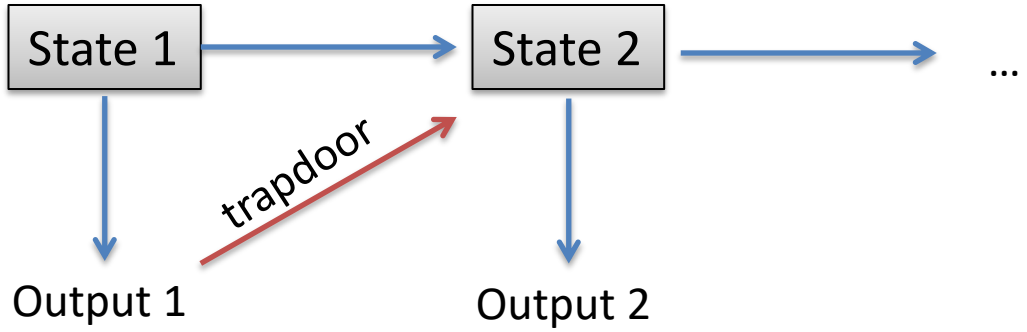
(S3, R2) <- PRG(S2)

⋮



Sabotaging PRNGs

- NIST's Dual EC pseudorandom number generator (PRNG) apparently backdoored
  - Mandated public parameters are public key
  - There exists a secret key, the trapdoor (known since 2005 Shumow, Ferguson)
- One output of PRNG + trapdoor reveals next state of PRNG, and prediction of future outputs



A Simple Diffie-Hellman Trapdoor

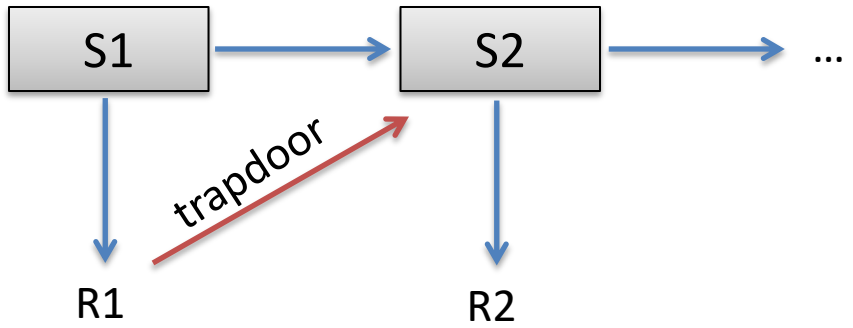
Let G be a cryptographically strong group with generator g

Let P in G be chosen parameter. Choose to be $P = g^p$

Let seed S1 be uniform value in $\mathbf{Z}_{|G|}$

$$PRG(S1) = (H(P^{S1}) , g^{S1}) = (S2,R1)$$

Given R1, p, compute $S2 = H(R1^p)$



Can view R1 as public-key encryption of next seed S2

Good PRNG to anyone without trapdoor p

Dual EC is very similar

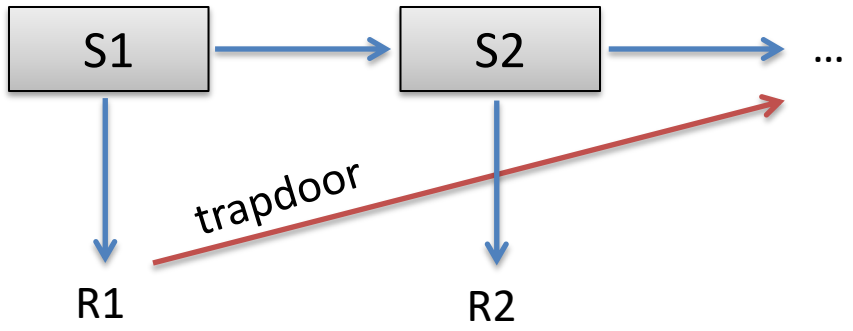
Let G be a cryptographically strong group with generator g

Let P in G be chosen parameter. Choose to be $P = g^p$

Let seed S1 be uniform value in $\mathbf{Z}_{|G|}$

$$PRG(S1) = (P^{S1} , g^{S2}) = (S2,R1)$$
$$PRG(S2) = (P^{S2} , g^{S3}) = (S3,R2)$$

Given R1, p, compute $S3 = R1^p = g^{S2*p} = P^{S2}$

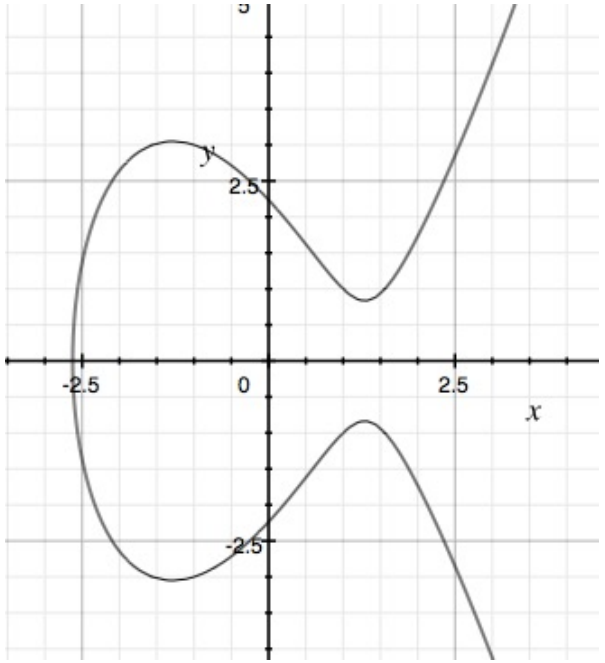


Actually, truncates 16 bits from R1. Can brute-force

Dual EC in papers

EC stands for elliptic curve: set of (x,y) points that are solutions to polynomial equation over finite field (plus point at infinity). Group operation is point addition

Point P on curve is x, y pair. x( ) returns x coordinate

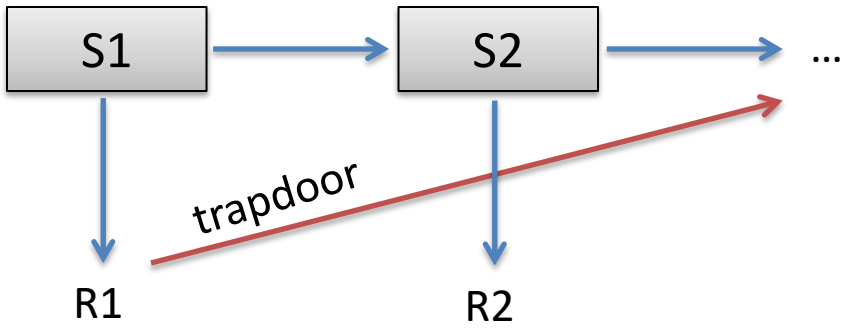


$y^2 = x^3 - 5x + 5$

(over the reals)

PRG(S1) = (x(S1*P) , x(S2*Q)) = (S2,R1)

PRG(S2) = (x(S2*P) , x(S3*Q)) = (S3,R2)



Must know $d = \log_Q P$. Either choose P = d*Q or Q = 1/d*P

How easy is Dual EC to exploit in practice?

- PRNG may not be used in exploitable ways
  - May not be used in first place (many faster PRNGs out there)
  - More bits of R1 may be truncated
  - May be implemented incorrectly
  - Dual EC supports *additional inputs* that could add entropy to each derivation, making attacks harder

Checkoway et al. 2014 study

- Investigate implementations of TLS:

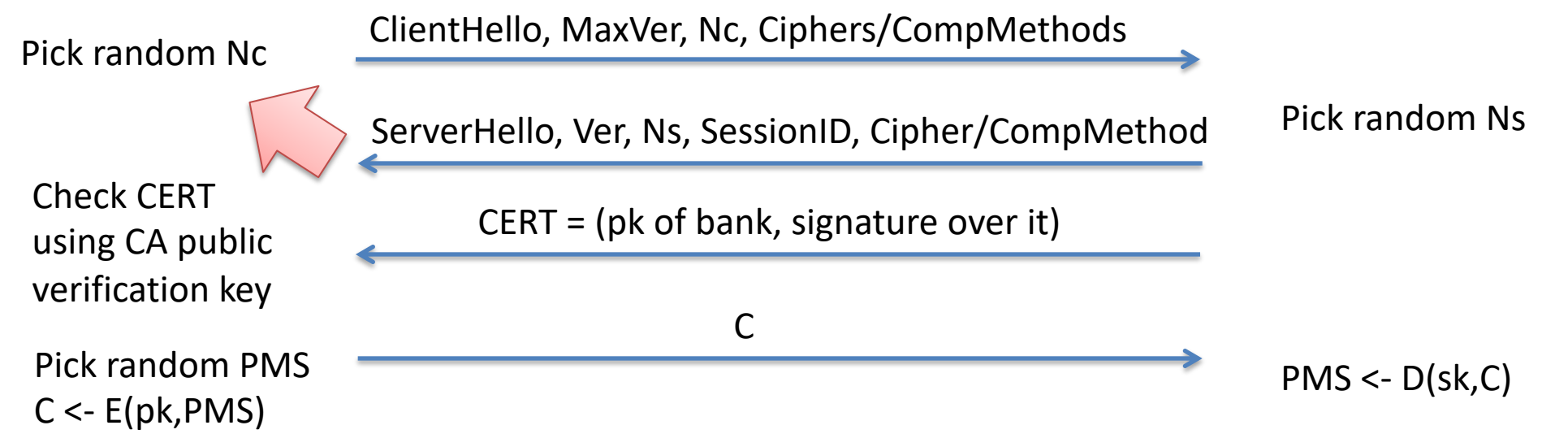
  openssl, Windows schannel, RSA BSAFE

- Conclude that some are more vulnerable than others:
  - Openssl bug prevents use of Dual EC (easy to fix)
  - Windows schannel uses additional input (deviates from Dual EC spec in ways that make attack faster)
  - RSA BSAFE very vulnerable

TLS handshake for RSA transport

Bank customer

Bank



Say client is using Dual EC for randomness generation

What is vulnerable?

RSA BSAFE library: 2.4 seconds to recover PMS

Windows: 60 minutes

OpenSSL: never (bug in code!)

Checkoway et al. 2014 study

Library	Default PRNG	Extended Random	Bytes per Session	Additional Entropy	Time (minutes)
BSAFE C	✓		31–60	—	0.04
BSAFE Java	✓	✓	28	—	63.96
SChannel I			28	—	62.97
SChannel II			30	—	182.64
OpenSSL-fixed I			32	20	0.02
OpenSSL-fixed II			32	35	83.32
OpenSSL-fixed III			32	$35+k$	$2^k \cdot 83.32$

ZMap scan of IPv4: only 720 servers using BSAFE Java

Juniper Dual EC Incident

[Checkoway et al. 2016]

- ScreenOS used in Juniper NetScreen firewall products. Used to perform VPN encryption (IPsec)
- Uses Dual EC, but supposedly wrapped within another PRNG. Shouldn’t be vulnerable, even to someone with trapdoor
- But it was. Worse, someone broke in and modified Q to a new value Q’
- Single 2008 patch modified P, introduced bug disabling secondary PRNG

Reversed ScreenOS PRNG

- Bunch of work to reverse engineer code
- Discover subtle bug in logic that disabled second PRNG step, thereby outputting Dual EC outputs directly to caller

- Really confusing history, summed up:
  - Juniper built backdoor into their code, Dual EC + subtle bugs + configuration of IKE
  - Someone broke in to Juniper and replaced Q with malicious Q' in 2012

Listing 1: The core ScreenOS 6.2 PRNG subroutines.

```
1  void prng_reseed(void) {
2    blocks_generated_since_reseed = 0;
3    if (dualec_generate(prng_temporary, 32) != 32)
4      error_handler("FIPS ERROR: PRNG failure, "
5                    "unable to reseed\n", 11);
6    memcpy(prng_seed, prng_temporary, 8);
7    prng_output_index = 8;
8    memcpy(prng_key,
9           &prng_temporary[prng_output_index], 24);
10   prng_output_index = 32;
11 }
12
13 void prng_generate(void) {
14   int time[2];
15   time[0] = 0;
16   time[1] = get_cycles();
17   prng_output_index = 0;
18   ++blocks_generated_since_reseed;
19   if (!one_stage_rng())
20     prng_reseed();
21   for (; prng_output_index <= 31;
22        prng_output_index += 8) {
23     // FIPS checks removed for clarity
24     x9_31_generate_block(time, prng_seed, prng_key,
25                          prng_block);
26     // FIPS checks removed for clarity
27     memcpy(&prng_temporary[prng_output_index],
28            prng_block, 8);
29   }
30 }
```

Policy

- “Going dark” debate over last few years
  - Police and others argue encryption is preventing criminals from being caught
  - Push for building in backdoors into crypto & other systems
  - Manhattan DA have interesting report about smartphone unlocking
- Cryptographers & security folks arguing that mandated backdoors are really bad idea
  - Keys under doormats report

Academic computer security research

- Researchers play active role in uncovering state of computer security in the world, and understanding adversarial actions
- Useful in informing debates like encryption policy