



SECURE COMMUNICATIONS AT THE SPEED OF CYBER

# Information Leakage Background and Threat Modeling

Suresh C. Kothari

Richardson Professor

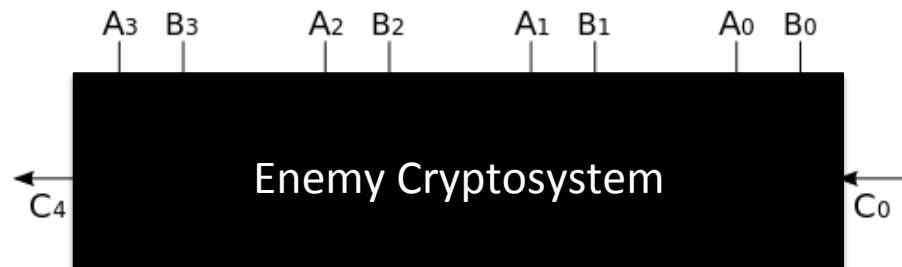
Department of Electrical and Computer Engineering

Ben Holland, Iowa State University

Acknowledgement: Team members at Iowa State University and EnSoft, DARPA contracts FA8750-12-2-0126 & FA8750-15-2-0080

BALTIMORE, MD • NOVEMBER 1–3, 2016

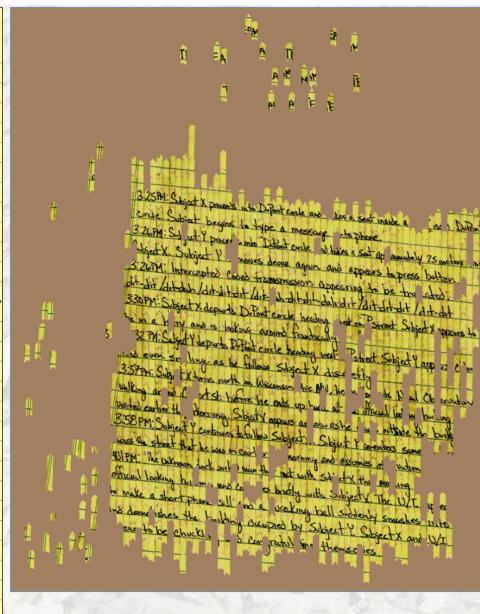
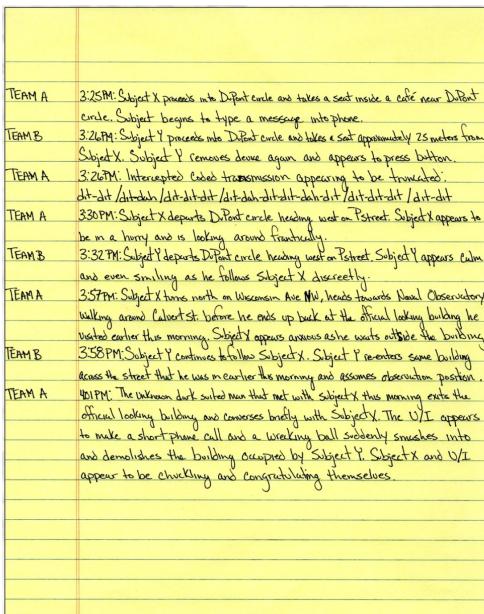
- Historically “side channels” were used to describe attacks to physical crypto hardware systems
  - Power analysis
  - Timing information
  - Acoustics
  - Faults
  - Electromagnetic radiation
    - Light, heat, IR, etc.
  - Not brute force/cryptanalysis
- Some operations require more time, power, etc. to complete than others



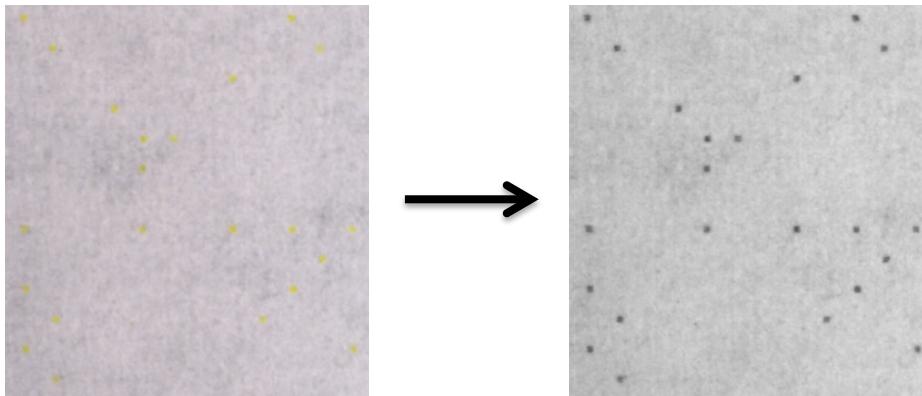
$A_0$	$B_0$	$A_1$	$B_1$	$A_2$	$B_2$	$A_3$	$B_3$	$C_0$
0	<b>1</b>	<b>1</b>	<b>0</b>	1	0	0	0	0
+	<b>0</b>	<b>1</b>	<b>1</b>	-	$S_0$	$S_1$	$S_2$	$S_3$
(1)	0	0	1	1	0	0	1	0

## DARPA's Paper Shredder Challenge

- \$50,000 prize to unscramble 5 shredded documents
- Puzzles were completely solved on December 2011 by team “All Your Shreds Are Belong To U.S.”



- A little of life's irony...
  - ~9000 teams competed, 1 team solved all 5 puzzles
  - Solution used hidden printer dots added by printer manufacturers and U.S. Secret Service



- Vision recognition software detected dots printed on paper and used dots as a reference guide to identify document fragments
- Pro-tip: Burn your documents you really want gone...

## ACHTUNG!

Alles turisten und non-teknischen  
looken peepers! Das computermaschine  
ist nicht für gefingerpoken und  
mittengraben! Ist easy schnappen der  
springenwerk, blowenfusen und  
poppencorken mit spitzensparken.

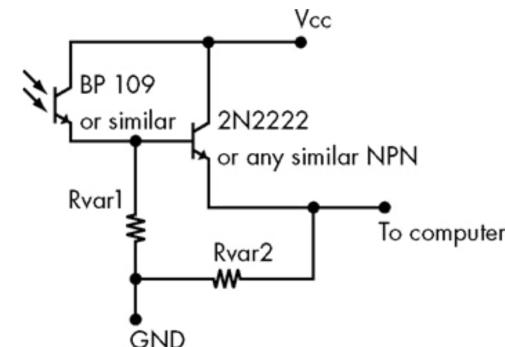
Ist nicht für gewerken bei  
dummkopfen. Das rubbernecken  
sightseeren keepen das cotton-picken  
hans in das pockets muss; Zo relaxen  
und watschen der blinkenlichten.



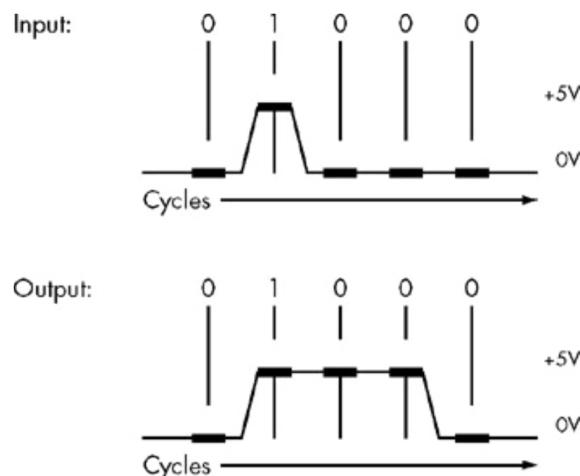
Historically, the blinking lights indicated important things like the state of the system, but as computers became faster and more reliable the lights were either removed or left as diagnostic indicators (example: networking hardware).

- LEDs on/off time is very fast (almost instant)
  - LEDs are usually used to control fiber optics
- LEDs were wired directly into the serial data line
  - Each blink is a 1 on your network, LED off is a 0
  - Too fast for a human eye
  - Not too fast for a circuit...and a telephoto lens!

Paper: Joe Loughry and David A. Umphress.  
2002. *Information leakage from optical emanations*. ACM Trans. Inf. Syst. Secur. 5, 3 (August 2002), 262–289.



- Duct tape over the LEDs works, but we still want our blinkenlights!
  - Pulse Stretching ☺



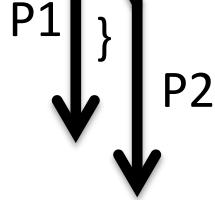
Still possible to recover 99.999988% of bits.  
Error correction codes can help us guess the rest.

- Better approach → Low frequency sampling with a latch till the next sample

- Short story: optimizations
  - Reducing cost: power, heat, etc.
  - Increasing speed/efficiency
- Consider synchronous vs. asynchronous digital logic circuits
  - Synchronous circuits operate on a fixed clock, all operations take the same time, so the best case and worst case times are the same
    - Every case is the worst case
  - Asynchronous circuits operate without a clock independent of other modules, so there are distinct best, worst, and average cases.
    - Average case costs less than the worst case

# Differential Branching

```
if(secret){  
    doLongAction();  
}
```



Path 1 Cost: Short

Path 2 Cost: Long

How many interactions with the *entrypoint* does it take to learn the *CONFIDENTIAL* value?

```
1 public class SideChannelExample {  
2     private static final int CONFIDENTIAL = 666;  
3     public static boolean entrypoint(int param){  
4         int condition = CONFIDENTIAL & param;  
5         if(condition == 0){  
6             for(int i=0; i<5; i++){  
7                 doSomethingExpensive();  
8             }  
9         } else{  
10             doSomethingCheap();  
11         }  
12         return CONFIDENTIAL == param;  
13     }  
14     private static void doSomethingCheap(){}
15     private static void doSomethingExpensive(){
16         for(int i=0; i<10000; i++){}  
17     }  
18 }
```

- Software information leakage primarily through...
  - Timing information
  - Memory space usage
    - Content, order, size
-  Space/Time usage are related problems
- Optimizations create asymmetries everywhere...
  - Software algorithms
    - Branching, short-circuiting logic, looping, etc.
  - Compiler optimizations
  - Cache hits
  - Process scheduling
  - Branch prediction...and so on...

## Demasking Google Users

1. Select Google users to target
2. Create a Google drive document and invite targets  
(uncheck option to send notification)
3. Using HTML/JavaScript create a spear-phishing site  
that identifies and customizes itself for the target
  - <img src=https://docs.google.com/document/...../edit />  
takes longer to call onerror if visitor is a target
  - Google has declined to fix this issue

# CRIME Vulnerability

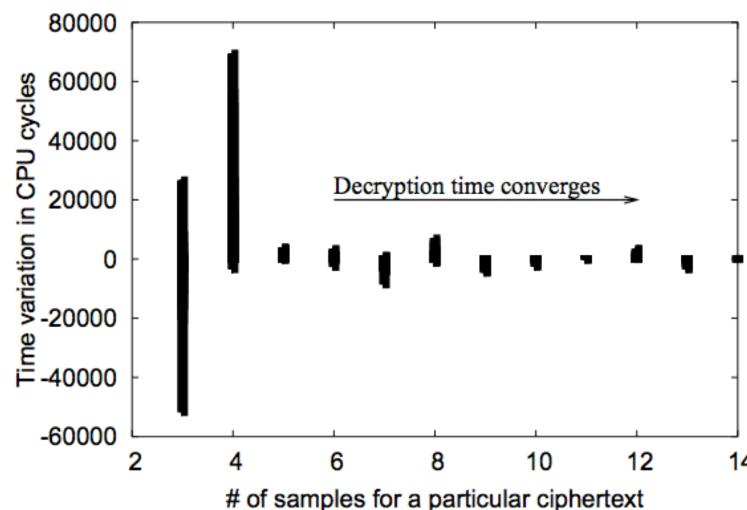
- HTTP Cookie Header can be extracted via size of compressed request

- Timing attack against OpenSSL server to recover SSL private RSA key
  - RSA decryption:  $m = c^d \bmod N$ , where  $N = pq$
  - If you know the factorization of  $N$ , then  $d = e^{-1} \bmod (p - 1)(q - 1)$
- Issue: Algorithm processing time was dependent on ciphertext and private key
  - Extra reductions in a Montgomery reduction (fast mod operation) when ciphertext ( $c$ ) approaches a multiple of  $q$  ( $c < q$  should be slower than decryption of  $c > q$ )
  - OpenSSL uses two different multiplication routines: when  $c < q$  fast Karatsuba multiplication is used, otherwise the slower normal multiplication is used since  $c > q$  is likely smaller when computing  $c \bmod q$
- Performed over a network (dealing with network delays)

- Use of repeated requests to recover modulus  $N$  of the public key
  - Binary search of most significant bits
  - After half the bits are recovered factorization is completed with Coppersmith's algorithm

Note that not all of the secret was leaked!

Just enough of the secret was leaked to make brute force search feasible.



- Good example of a malicious side channel
  - Source has to be capable of passing a peer review
  - Execution has to appear to perform the task correctly
  - Challenge: Censor regions on a JPEG image, but somehow leak the redacted information
- Snippet of winning solution:

```
//read the ppm header
unsigned width, height, maxdepth;
fscanf(ppm, "P3\n%u %u\n%u\n", &width, &height, &maxdepth);
printf("P3\n%u %u\n%u\n", width, height, maxdepth);
```

- Writes the magnitude of the red, green, and blue component for each pixel in order

- When we censor with a black rectangle we write 0's for the RGB pixel (a black pixel)
- PPM file format is flexible and implementation leaks how many digits each value originally when it processes the file character by character

234 2 0 83 255 255 2 43 255

Implementation: 000 0 0 00 000 000 0 00 000

Should write: 0 0 0 0 0 0 0 0 0 0

- Perfect reconstruction for black/white images, otherwise partial reconstruction of blacked out region

## Thoughts on Noise

- Example password segmented side channel
  - Signal degradation
- What is the strength of the side channel?

# Threat Modeling

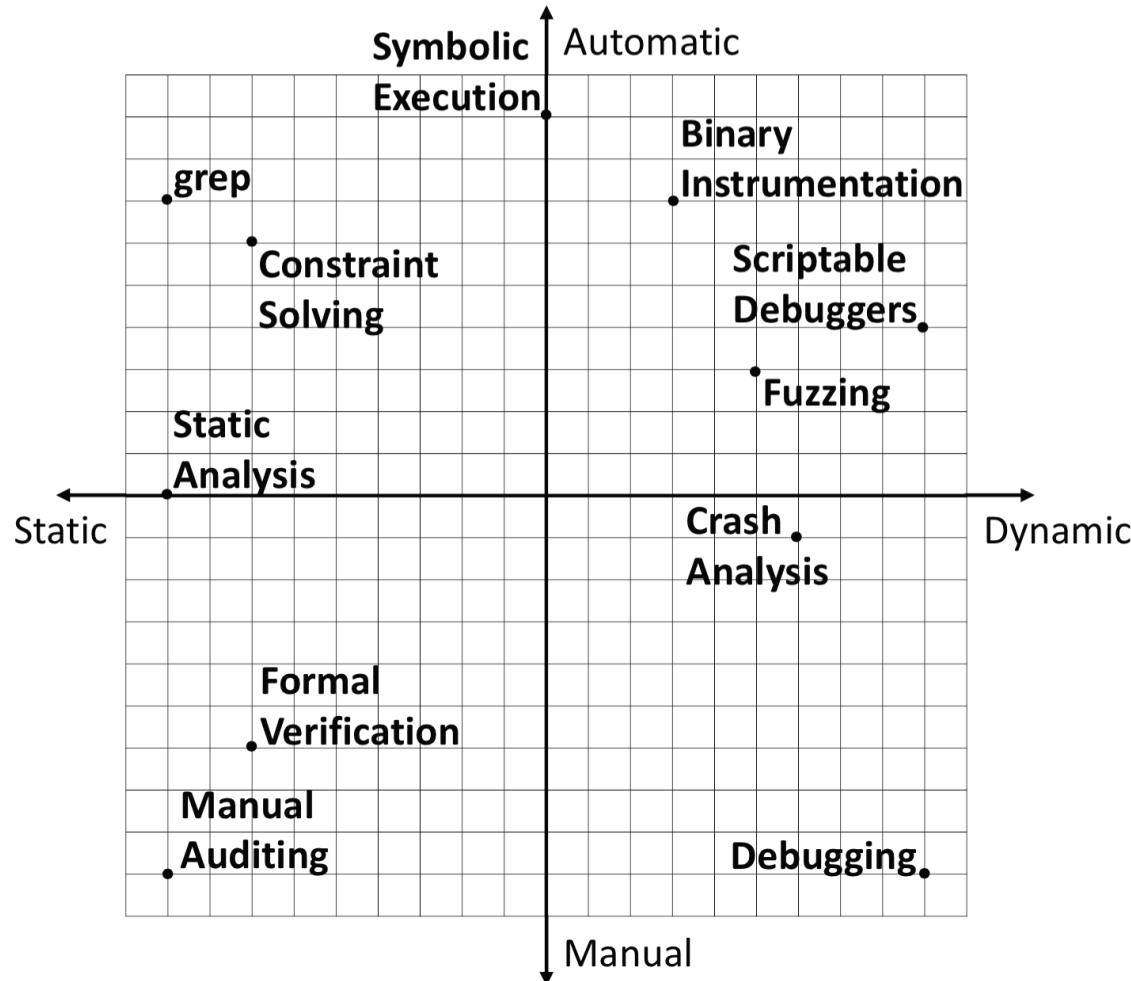
- Budget in terms of active/passive operations?
- Threshold for # interactions?
- What are the observables?

# References

- [1] DARPA Paper Shredder Challenge – <http://archive.darpa.mil/shredderchallenge>
- [2] U.S. Secret Service Printer Program – <http://seeingyellow.com>
- [3] Blinkenlights (Chapter 5) – Michal Zalewski. 2005. *Silence on the Wire: A Field Guide to Passive Reconnaissance and Indirect Attacks*. No Starch Press, San Francisco, CA, USA.
- [4] [Demasking Google Users with a timing attack](#). Andrew Cantino. 2014.
- [5] OpenSSL Timing Attack – Brumley, David, and Dan Boneh. [Remote timing attacks are practical](#). Computer Networks 48.5 (2005): 701-716.
- [6] Underhanded C Contest – <http://notanumber.net/...the-leaky-redaction>

# Recommended Reading

- [1] [Eliminating Timing Side-Channels. A Tutorial.](#) Peter Schawabe. ShmooCon 2015.
- [2] [Remote timing attacks are practical.](#) Brumley, David, and Dan Boneh. Computer Networks 48.5 (2005): 701-716.
- [3] [Side Channel Attacks.](#) John Franco. University of Cincinnati Network Security course lecture.
- [4] [Silence on the Wire: A Field Guide to Passive Reconnaissance and Indirect Attacks.](#) No Starch Press, San Francisco, CA, USA. Michal Zalewski. 2005.



Source: Contemporary Automatic Program Analysis,  
Julian Cohen, Blackhat 2014