Additional Hardware Techniques

- Implement a Memory Protection Unit (MPU)
  - Run “application” code in user mode and the “kernel” in supervisor mode
  - Only supervisor mode has access to the MPU
  - Prohibits “rogue” application code from accessing memory of other processes
- Recently in the news Intel’s Spectre and Meltdown exploits, see this link

You can have a memory protection unit in there,

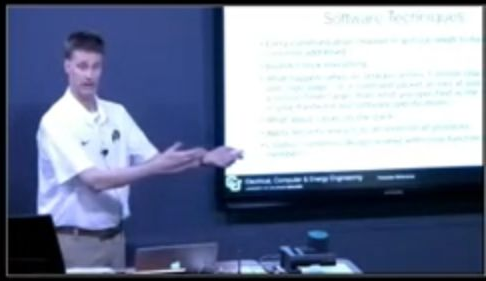
Other Protection Methods

- Tamper evidence
  - Sticker
  - Epoxy resin (conformal coating), deterrent, but difficult for returned unit failure analysis
- Proximity detection, enclosure breach

That is something you can do whether it makes

Software Techniques

- Every communication channel in and out needs to have security concerns addressed.
- Bounds check everything
- What happens when an attacker enters 1 million characters into a web login page - or a command packet arrives at your device that is a million times larger than what you specified as the maximum size in your hardware and software specifications?
- What about values on the stack?
- Apply security analysis to all levels of all protocols
- Conduct numerous design reviews with cross-functional team members



You get your product out the door faster.

Other Threat Vectors?

- Can you think of any unanticipated access channels an attacker might use?
  - Power
  - RF
  - Scan-chains
  - JTAG port
  - Temperature
  - ?

There might be others. So we left off here last time,

Side Channel Attacks

- Differential power analysis
- RF analysis

and they had an app on their iPhone that was running AES 256 encryption algorithm,

Side Channel Attacks

- Differential power analysis
- RF analysis
- Power glitching
  - Puts hardware into an unknown and potentially compromised state
- Radiation
- See: Cryptography Research (a division of Rambus) for more information

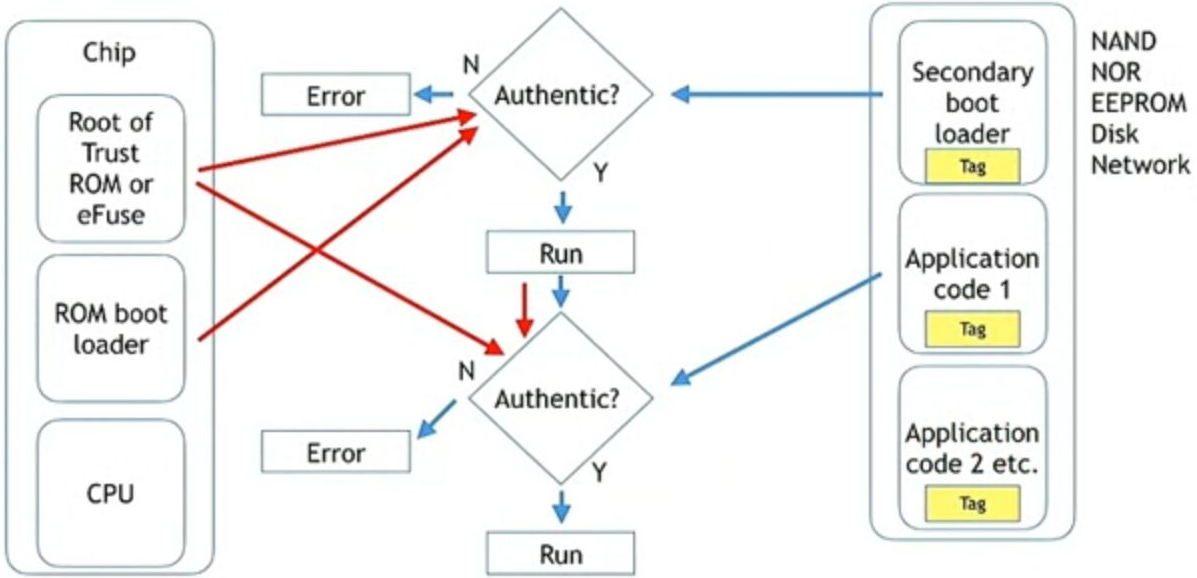
The RF analysis one was pretty cool.

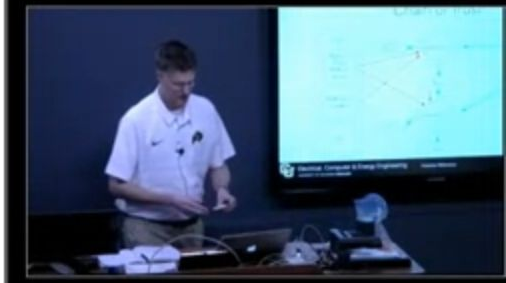
Additional Info

- Cloud Security Alliance
- https://cloudsecurityalliance.org/download/security-as-a-service-working-group-charter/
- Bloomberg's 'What is Code?' article
  - https://www.bloomberg.com/company/announcements/bloomberg-businessweek-releases-code-issue-special-multi-platform-package-demystifying-code/
- Wall Street Journal
  - http://partners.wsj.com/bitdefender/history-of-hacking/watch-evolution-cyber-crime/

This is the history of hacking and the evolution of cyber crime.

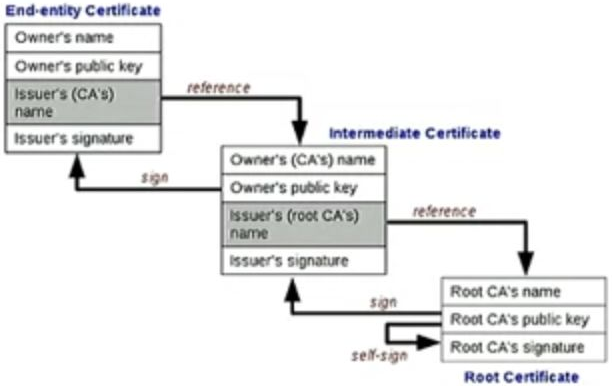
Chain of Trust





The next subject I want to talk about is Chain of Trust.

Chain of Trust (con't)



Source: https://en.wikipedia.org/wiki/Chain_of_trust

A generic Secure Boot architecture is pictured.

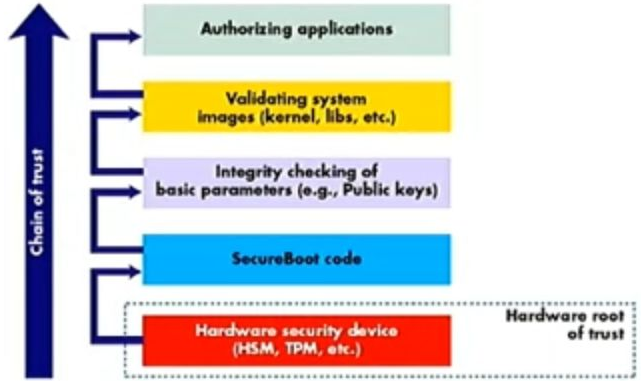


Figure 2

Source: https://www.embedded.com/design/prototyping-and-development/4007195/Employ-a-secure-flavor-of-Linux

but again it shows how you can start with a Root certificate that is authentic.

Onity HT Door Lock

Source: Cody Brocious http://demoseen.com/bhpaper.html

So, there's a company called Onity that made

Onity HT Door Lock - Data on a card key

16-bit ident value
Value assigned identifies which lock the card is associated with.
Employee master key = employeeID.

8-bit flags byte
Used for misc options.

16-bit expiration date
Defines length of time card is valid.

24-bit unknown field
Set to all 0's.

24-bit keycode value + look-ahead value
Locks are programmed with these values, ex: 100 and 50, lock would accept cards with values 100 to 150.
Every time a valid card is inserted, the lock resets its keycode value to the keycode value from the card, thereby invalidating older cards.
Keycodes representing master keys are also programmed into the locks.

The data fields on that track consists of the following fields,

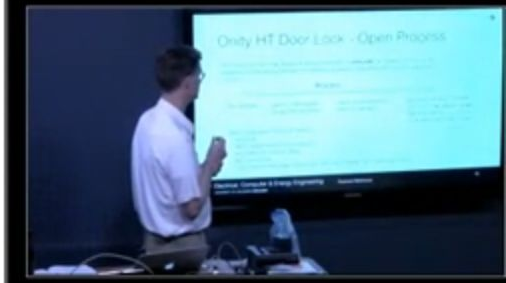
Onity HT Door Lock - Open Process

The fields on the mag strip are encrypted with a **sitecode**, a random 32-bit value, assigned by the manufacturer to identify a specific property. Encryption algorithm is custom.

Process

Card swiped	Data is decrypted using the sitecode	ident and expiration date is checked	keycode value is checked and if in look-ahead range the lock opens. Lock saves keycode value

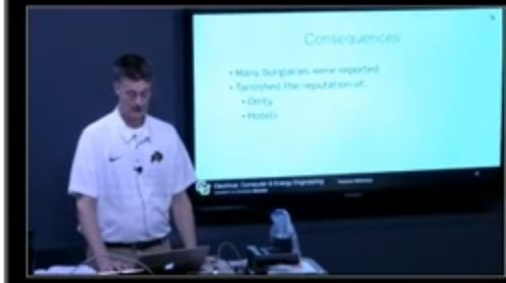
What data does the lock have?
- sitecode
- ident value (which lock am I?)
- keycodes for master keys
- date and time
- most recent keycode value from last card swipe/lock open operation



The fields on that mag strip are encrypted with a site code,

Consequences

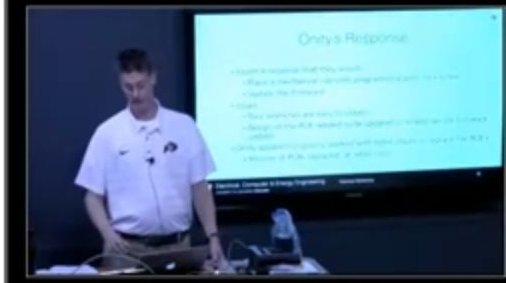
- Many burglaries were reported
- Tarnished the reputation of:
  - Onity
  - Hotels



This was a big deal.

Onity's Response

- Issued a response that they would:
  - Place a mechanical cap over programming port, torx screw
  - Update the firmware
- Issues:
  - Torx wrenches are easy to obtain
  - Design of the PCB needed to be updated to enable secure firmware update
- Onity apparently quietly worked with hotel chains to replace the PCB's
  - Millions of PCBs replaced, at what cost?



These boo boos can cause companies a huge amount of money,

Z-Wave Door Lock

Researcher's controller

Z-Wave controller

1. Device is unpaired, initial key exchange is performed
2. Common key determined, stored in EEPROM, device now paired
3. *Frame encryption* key generated, used to encrypt payloads in subsequent communications
4. *Data origin authentication* key generated used to generate a MAC value to address replay attacks

The flaw:
The researchers could transmit a new key-exchange packet.
The **lock firmware** failed to check if there was already an existing common key in EEPROM,
and so went through the key exchange sequence again, enabling an attacker to pair and unlock the lock.

Source: Behrang Fouladi, Sahand Ghanoun

so this is the researcher's Z-Wave controller over here.

Bluetooth Vulnerabilities

Master and slave can use encryption to secure data exchange.
Must establish a shared secret known as the Long-Term-Key (LTK).

The exchange process starts by selecting a temporary key (TK).
According to the BLE specification, TK=0 if "Just Works" mode is selected.
Just Works mode is used for devices with no display/input capability. Otherwise a TK value from 0 to 999999 is used.
Once the TK is established, the devices establish a Short-Term-Key (STK), and eventually establish a LTK.

Some Bluetooth master and a slave can use encryption to secure

Bluetooth Vulnerabilities

Ryan created a tool called Crackle.
Ryan captured the BLE data exchange and input the data to crackle.
Crackle attempts to brute-force the TK by choosing values from 0 to 999999.
Once the TK is found, the STK can be found by decrypting it with the TK. Then the LTK can be found by decrypting it with the STK.

The flaw:
The range of TK's is relatively small. **In this case it was practical to try every key.**

Devices don't have to use the "Just Works" BLE specification and can rely on schemes where the keys are 128-bits.

Problem with that is Ryan created a tool called Crackle,

Incidents In The News

Sony's Settlement With Employees Over Hacked Data Worth More Than $5.5 Million

Critical Unpatched Flaws Disclosed In Western Digital 'My Cloud' Storage Devices

tom's HARDWARE

SECURITY > NEWS

Intel AMT Allows BitLocker Bypass In Under A Minute

41 percent of Android phones are vulnerable to 'devastating' Wi-Fi attack

Every Wi-Fi device affected by some variant of attack

Windows Central

Lenovo discloses security vulnerability in ThinkPad fingerprint manager

Lenovo's fingerprint management software for Windows 7 and 8 has a pretty major vulnerability but a patch is available

Seagate Quietly Patches Dangerous Bug in NAS Devices

Samsung Left Millions Vulnerable to Hackers Because It Forgot to Renew a Domain, Researchers Say

The hackers who broke into Equifax exploited a flaw in open-source server software

CR Consumer Reports

Major Security Flaw Found in Netgear Routers

The Register

DATA CENTER SOFTWARE NETWORKS SECURITY INFRASTRUCTURE BUSINESS HARDWARE SCIENCE

Security

Western Digital's hard drive encryption is useless. Totally useless

Rookie errors make it child's play to decrypt data

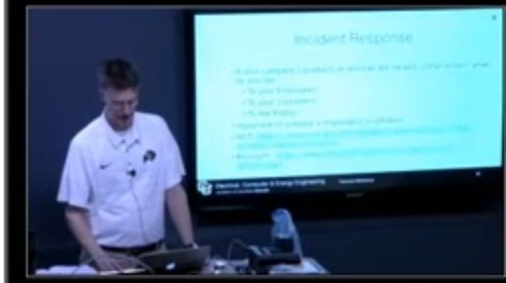
Security

Qualcomm joins Intel, Apple, Arm, AMD in confirming its CPUs suffer hack bugs, too

Target To Pay $10 Million To Settle Lawsuit From Massive Data Breach

Incident Response

- If your company's products or services are hacked/compromised, what do you say:
  - To your Employees?
  - To your Customers?
  - To the Public?
- Important to prepare a response(s) in advance
- NIST: https://www.nist.gov/el/intelligent-systems-division-73500/incident-response-scenarios
- Microsoft: https://www.microsoft.com/en-us/cybersecurity/default.aspx



incident reporting and Microsoft has a bunch of information here at this link as well.

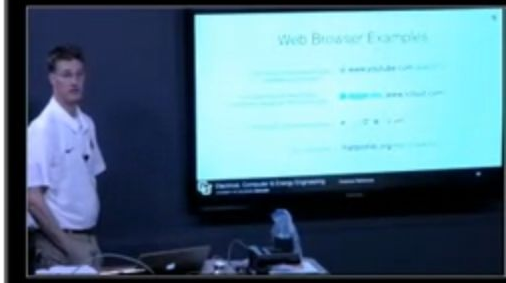
TLS / SSL

- Transport Layer Security
- Secure Socket Layer (predecessor to TLS)
- When a connection is secured by TLS, in our case a web browser and a server, the connection will have one or more of the following properties:
  - The connection is ***private*** (secure) via symmetric encryption
  - The identity of the communicating parties can be ***authenticated*** using public-key encryption, they are who they say they are
  - The connection has ***integrity*** because each message exchanged is protected by a MAC to detect alteration

it's not being modified while it's in flight.

Web Browser Examples

Encrypted and authenticated (standard certificate)	www.youtube.com/watch?v=
Encrypted and authenticated (extended validation (EV) certificate)	Apple Inc. www.icloud.com/
Encrypted, not authenticated	~~https~~://
Not encrypted	matplotlib.org/mpl_toolkits/r



So here's some examples of how web browsers tell us what the level

What is a Digital Signature and Certificate

ny that owns
b server
a certificate
nds to a CA to
ned"

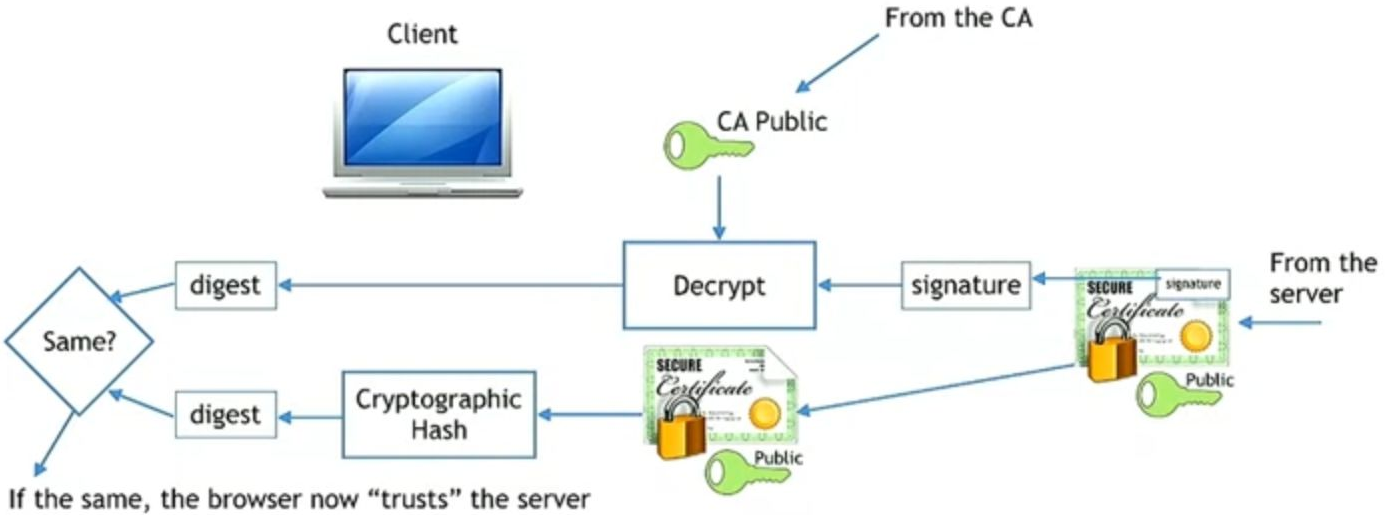
equesting company

y publicly available

://searchsecurity.techtarget.com/definition/digital-signature

and just think of a certificate

Step 3 - Authenticate Certificate



So, the client will receives the certificate with the signature and the public key,

Certificate Examples

com.apple.idms.appleid.prd.4c6b3932767546495845 4e366c627530624978635477 3d3d
Issued by: Apple Application Integration Certification Authority
Expires: Thursday, June 22, 2017 at 7:07:06 PM Mountain Daylight Time
This certificate is valid

▸ Trust
▾ Details

Subject Name
Common Name com.apple.idms.appleid.prd.4c6b393276754649584 64e366c627530624978635477 3d3d

Issuer Name
Country US
Organization Apple Inc.
Organizational Unit Apple Certification Authority
Common Name Apple Application Integration Certification Authority

Serial Number 7262256962065564017 25
Version 3

Signature Algorithm SHA-256 with RSA Encryption (1.2.840.113549.1.1.11)
Parameters none

Not Valid Before Tuesday, June 23, 2015 at 7:07:06 PM Mountain Daylight Time
Not Valid After Thursday, June 22, 2017 at 7:07:06 PM Mountain Daylight Time

Public Key Info
Algorithm RSA Encryption (1.2.840.113549.1.1.1)
Parameters none
Public Key 256 bytes : F5 AC 80 24 45 65 A2 86 8A 57 2C B2 D1 1A 61 97 92 42 6E 24 68 08 1E FF B3 C5 88 3A 61 A8 38 E0 32 37 CB 60 13 71 25 EA F0 CC D7 F5 36 1C 5F 44 6E C6 52 69 AC 75 49 C5 96 1D F5 98 99 FD 3F 8C CD 52 94 29 C3 6F D2 94 6B 37 22 D8 B0 69 83 D4 D3 69 66 07 46 65 23 8A DC A8 C8 32 FA D1 CF 95 79 5D 5F 79 3E 23 B8 B5 63 3D 36 46 E0 5C AE A2 0E F8 95 6B 90 74 36 1D 77 1B C7 6C 04 27 74 2A 8F 0D 7F 5E F8 2C A0 0F 1B A1 6F 44 09 F5 32 51 34 B8 15 1D 6B 8E CA 07 87 4F 12 74 48 38 75 7F 9C 84 E8 B4 84 7E E1 38 17 13 63 DE 13 B8 6F 39 B0 2E E0 45 58 36 54 E7 51 41 A5 C8 D4 A7 D9 D3 B3 00 76 5F 46 C5 B1 03 D9 B0 4E 8F 44 93 A9 70 A3 B2 68 7F 5D 27 56 7B 81 D0 03 F8 F6 2D 91 BA F9 95 09 07 36 4F 33 BB D0 91 15 A9 F8 C7 30 86 61 38 DC DF A9 6F F2 A7 94 25 71 28 40 62 5F FF
Exponent 65537
Key Size 2048 bits
Key Usage Encrypt, Verify, Derive

Signature 256 bytes : A6 F1 64 0A A0 4C 96 82 08 59 3C 2F FB 24 35 37 68 AE 27 D3 3F 78 F5 2F B1 0E 3C 05 14 50 37 C4 5E 4B 53 BE 30 42 B8 36 00 CA 82 2A 37 01 C3 C6 B5 05 6C 6F 8E 98 F0 02 F5 01 74 40 21 AE 1C 18 44 C4 D4 39 95 1F C5 73 30 F2 E5 49 B6 72 40 40 3F 94 EE 71 8D 05 66 63 A9 4A 24 12 6F E0 6F B8 F2 DE 43 D1 3F 78 1A 7F BC 14 15 37 53 DF C0 98 33 2E FC 50 AE 27 E0 19 12 62 F1 76 64 6C DC 09 54 A2 06 22 01 FF 2D 9D 28 A2 B6 3A 8D 25 92 61 3D 70 FF 62 F3 F0 98 29 97 DF 65 B5 FB 1B C5 50 C8 5B 01 A2 0A 7E A9 44 90 39 3E 1B 55 09 19 A5 A2 65 17 8F 68 07 8D 2A F4 F5 2D 6D 2D F8 14 CE 98 FC AD E5 B3 A6 8D A1 29 72 59 EA 00 C2 B7 77 48 57 34 27 D9 85 6F 2B 77 75 A9 57 8E 04 B0 8B 20 87 37 A3 C6 F8 D8 3F 02 E9 05 EC 89 89 9B 68 A0 2D 8D F4 00 1A 59 C6 2A EA DE E3 93 E2 B4 8F

Lauterbach MAIL CA
Intermediate certificate authority
Expires: Sunday, October 11, 2020 at 7:37:47 AM Mountain Daylight Time
This certificate was signed by an unknown authority

▸ Trust
▾ Details

Subject Name
Country DE
State/Province Bavaria
Organization Lauterbach GmbH
Common Name Lauterbach MAIL CA

Issuer Name
Country DE
State/Province Bavaria
Locality Hoehenkirchen
Organization Lauterbach GmbH
Common Name Lauterbach ROOT CA

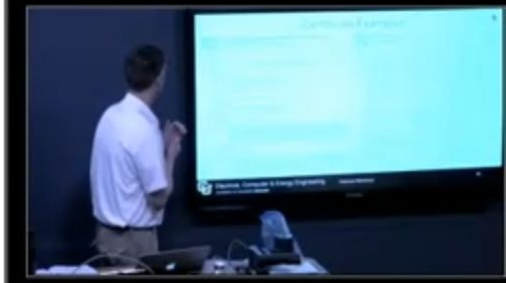
Serial Number 2
Version 3

Signature Algorithm SHA-1 with RSA Encryption (1.2.840.113549.1.1.5)
Parameters none

Not Valid Before Thursday, October 14, 2010 at 7:37:47 AM Mountain Daylight Time
Not Valid After Sunday, October 11, 2020 at 7:37:47 AM Mountain Daylight Time

Public Key Info
Algorithm RSA Encryption (1.2.840.113549.1.1.1)
Parameters none
Public Key 128 bytes : D3 38 D9 F8 AF 02 35 1A 7F 18 D1 B2 FF DB 31 5E 81 18 3E 85 09 22 46 18 F8 C5 B0 28 50 16 DF B0 1C 8F 58 3E 39 CD 2A A5 30 63 4B 40 ED 72 3B 52 7B 28 A6 E8 89 87 AD ED 6C 72 1E 52 C2 E3 DC 55 5F 63 6E 25 71 DA C6 ED 1D F6 F4 88 BC BE B0 F7 7B FE 81 ED D0 15 DF E3 4D 73 AB ED 81 AB A1 A5 63 03 22 85 82 0C 45 C8 EF AB EA 10 DF A5 A2 0E 3C B4 11 8A 6D 6A 3F B6 61 32 16 B9 35 DF C3 E7
Exponent 65537
Key Size 1024 bits
Key Usage Any

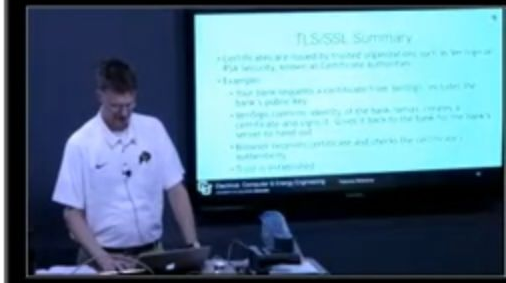
Signature 128 bytes : 88 45 BF AA 77 D3 13 B6 4D B5 62 07 BC 91 78 88 D4 B8 95 B6 6A 07 21 6B FF B4 DF 19 67 34 0F E2 C3 94 98 D2 F1 BC 41 1E 07 DC 51 D6 5D 65 E1 F0 37 60 3C 25 0F 06 05 74 1C B8 5F 90 C8 B9 67 65 A4 6B DE 0C 39 52 3B 1B CC 15 FD 03 7D 05 28 41 6A 57 78 92 24 6B CA 7E BE 4C 90 EB BC 0E 65 67 05 0E E0 5A D1 07 9F F0 D7 CA D2 4D C9 22 97 E6 77 A9 25 0A 6B 1E 31 76 01 65 59 DC 1D C6 3E 2B



This one was issued by Apple.

TLS/SSL Summary

- Certificates are issued by trusted organizations such as VeriSign or RSA Security, known as Certificate Authorities
- Example:
  - Your bank requests a certificate from VeriSign, includes the bank's public key
  - VeriSign confirms identity of the bank/server, creates a certificate and signs it. Gives it back to the bank for the bank's server to hand out
  - Browser receives certificate and checks the certificate's authenticity
  - Trust is established

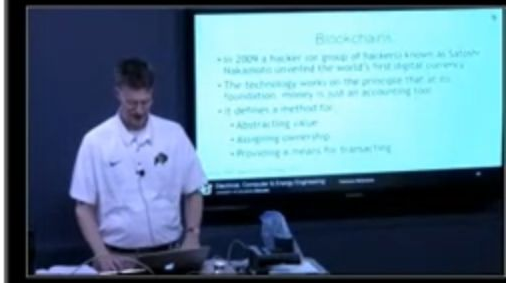


and checks the certificate's authenticity, and trust is established.

Blockchains

- In 2009 a hacker (or group of hackers) known as Satoshi Nakamoto unveiled the world's first digital currency
- The technology works on the principle that at its foundation, money is just an accounting tool
- It defines a method for:
  - Abstracting value
  - Assigning ownership
  - Providing a means for transacting

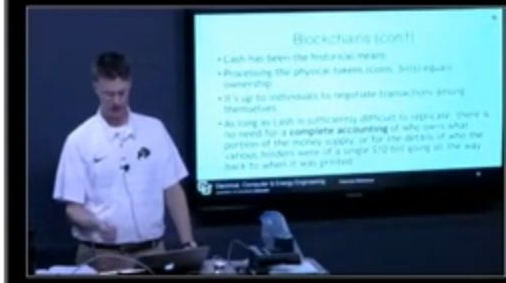
Source: IEEE Spectrum, October 2017



It provides a means for conducting transactions.

Blockchains (con't)

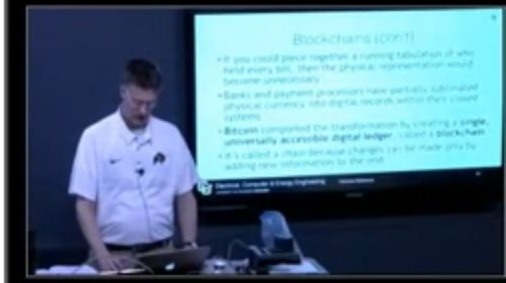
- Cash has been the historical means
- Processing the physical tokens (coins, bills) equals ownership
- It's up to individuals to negotiate transactions among themselves
- As long as cash is sufficiently difficult to replicate, there is no need for a **complete accounting** of who owns what portion of the money supply, or for the details of who the various holders were of a single $10 bill going all the way back to when it was printed



We don't have to have that accounting because whoever has it now

Blockchains (con't)

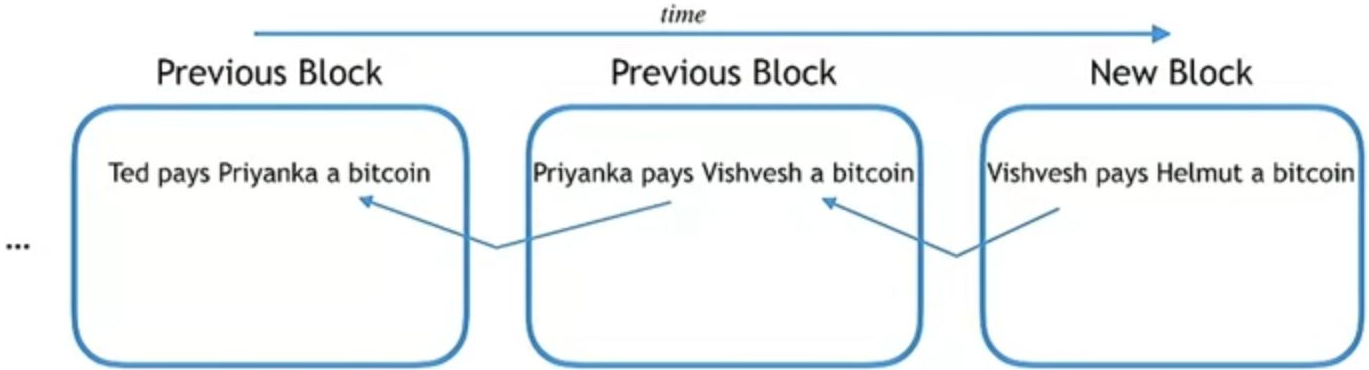
- If you could piece together a running tabulation of who held every bill, then the physical representation would become unnecessary
- Banks and payment processors have partially sublimated physical currency into digital records within their closed systems
- **Bitcoin** completed the transformation by creating a **single, universally accessible digital ledger**, called a **blockchain**
- It's called a chain because changes can be made only by adding new information to the end

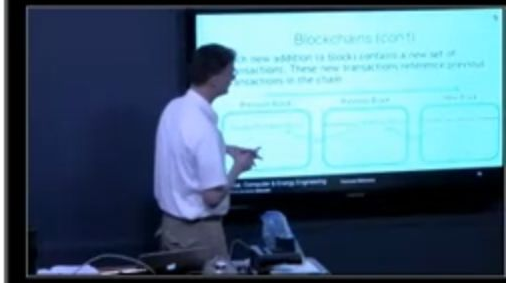


It's called a chain because the changes can only be made by adding

Blockchains (con't)

- Each new addition (a block) contains a new set of transactions. These new transactions reference previous transactions in the chain

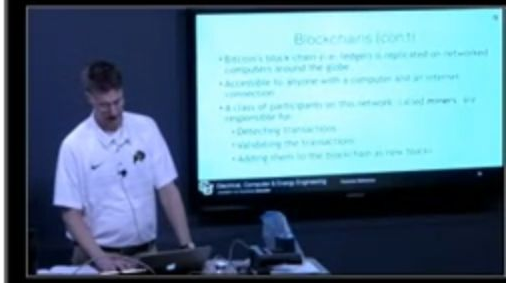




The reason this works is because it's computationally

Blockchains (con't)

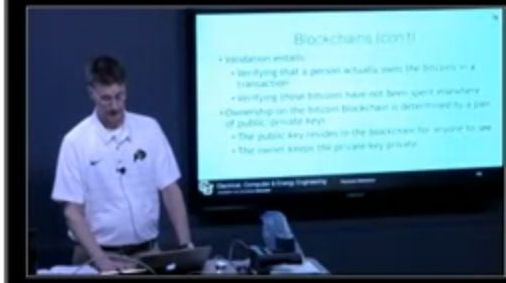
- Bitcoin's block chain (i.e. ledger) is replicated on networked computers around the globe
- Accessible to anyone with a computer and an internet connection
- A class of participants on this network, called **miners**, are responsible for:
  - Detecting transactions
  - Validating the transactions
  - Adding them to the blockchain as new blocks



validating those transactions, and adding them to the blockchain as new blocks.

Blockchains (con't)

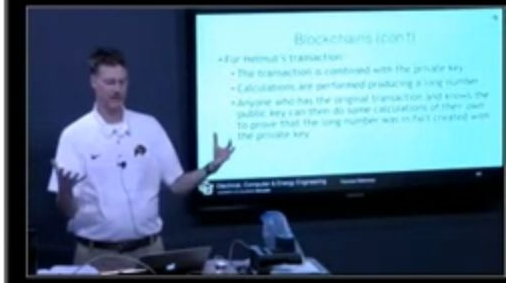
- Validation entails:
  - Verifying that a person actually owns the bitcoins in a transaction
  - Verifying those bitcoins have not been spent elsewhere
- Ownership on the bitcoin blockchain is determined by a pair of public/private keys
  - The public key resides in the blockchain for anyone to see
  - The owner keeps the private key private



the owner keeps the private key private.

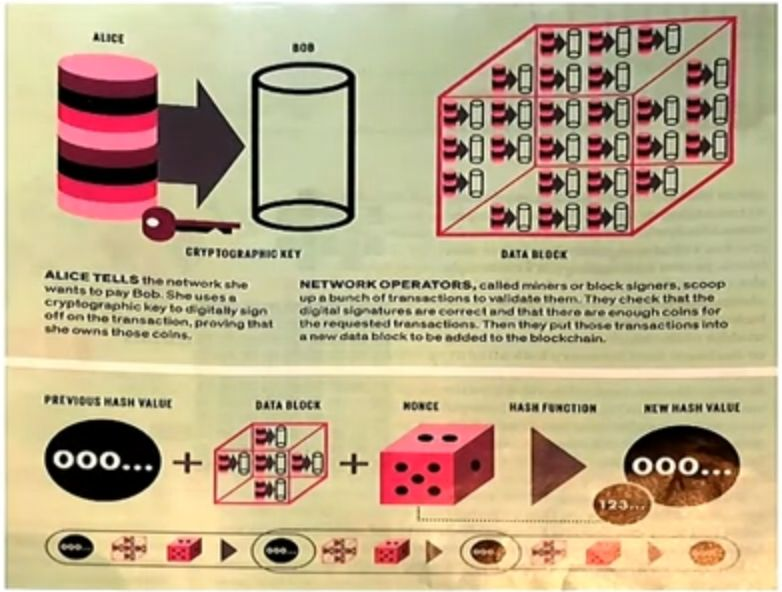
Blockchains (con't)

- For Helmut's transaction:
  - The transaction is combined with the private key
  - Calculations are performed producing a long number
  - Anyone who has the original transaction and knows the public key can then do some calculations of their own to prove that the long number was in fact created with the private key



and then the miner is paid in Bitcoins for finding that long number.

Blockchains (con't)



Network operators called miners or

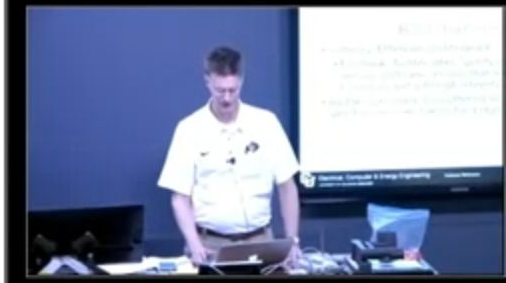
Blockchains (con't)

- Other uses for blockchains
- Ethereum
  - Unlike bitcoin, Ethereum uses miniprograms (called smart contracts) that can be written with unlimited complexity
  - Users can then interact with the miniprograms by sending them transactions loaded with instructions, which miners then process
  - What this means is that anyone can embed a software program into a transaction and know that it will remain there, unaltered and accessible for the life span of the blockchain

a transaction will know that it will remain

Blockchains (con't)

- In theory, Ethereum could replace
  - Facebook, Twitter, Uber, Spotify or any other digital service, with new versions that would be invulnerable to censors and with high integrity
- Another use: Initial Coin Offering (ICO) - think application specific coins, like tokens for a laundromat



the securities laws vary from country to country.

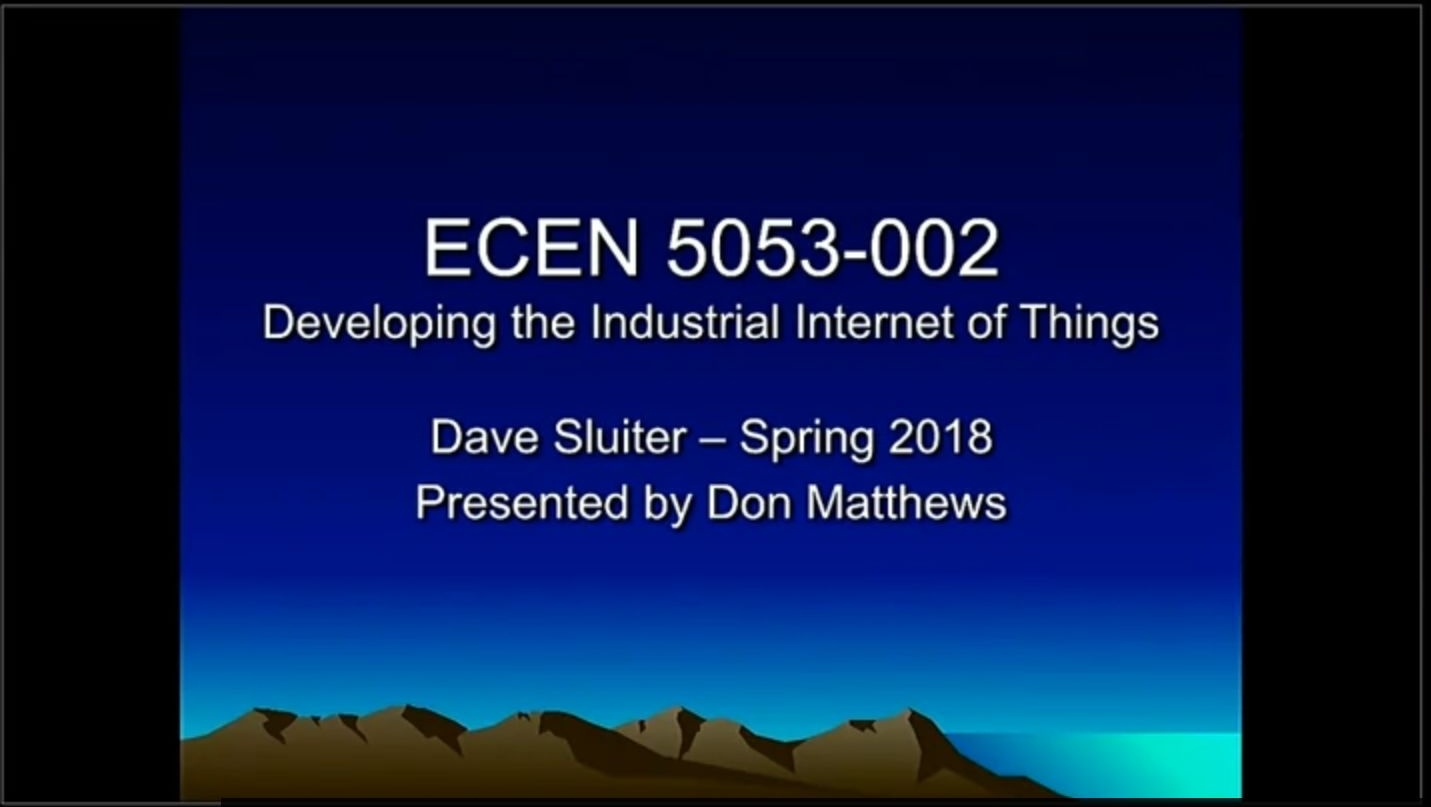
Blockchains (con't)

- Downsides:
  - Computing power consumption
  - Privacy laws
  - Each country has specific privacy laws: financial institutions, medical records

as a retarding force for the deployment of blockchains.

Blockchains (con't)

I didn't mark my source down here and should have been on the ball,



a past lecture we worked together at Seagate for many years,

Disclaimer

- All material and options presented by the author and speaker, Don Matthews, are expressly from the author and do not represent any opinions of any employer of the author.

12/12/2016 Don Matthews

If you know of some security failures let's talk about them.

Material

- What Algorithm/Protocols to use
- Anti-Tamper
- Threat Model
- Attacks
- Hard Drives
- Password Tables

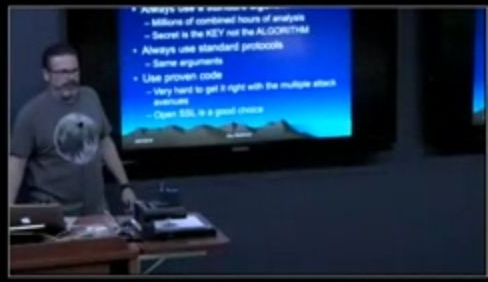
2/27/2016 Don Matthews

So, what algorithms do you want to use?

Algorithm/Protocol to Use

- Always use a standard algorithm
  - Millions of combined hours of analysis
  - Secret is the KEY not the ALGORITHM
- Always use standard protocols
  - Same arguments
- Use proven code
  - Very hard to get it right with the multiple attack avenues
  - Open SSL is a good choice

2/27/2016 Don Matthews



Standard algorithms have millions of combined hours people researching.

Attacks

- eBeam
  - Read out storage elements
  - Read the key or other critical values
- Focused Ion Beam (FIB)
  - Make changes to a chip circuit
  - Bypass security bits
- Light leakage
  - Observed stored values based on emitted light

2/27/2016 Don Matthews

How can I attack things? All right.

Attacks – Page 2

- Fault Injection
  - Power glitches, clock glitches, Low power, fast clocks
  - Force the chip to misbehave
    - Clock glitch when software checks an authentication value
  - Side Channel
    - Power usage (raw power, EM radiation)
    - Time

2/27/2016 Don Matthews

So, fault injection, and this is a whole study area by itself.

Attacks – Timing on RSA

- RSA: compute Y^X mod n
- Y, X, and n are 2, 3, or 4K bits in size (w)
- Series of Square operations and conditional multiply operations

- Let s0 – 1
- For k = 0 upto w-1
  - If (bit k of x) is 1 then
    - Let $R_k = (s_k * y) \bmod n$
  - Else
    - Let $R_k = s_k$
  - Let $s_{k+1} = R_k^2 \bmod n$
- End For
- Return (R_{w-1})

2/27/2016 Don Matthews

So, for RSA operations basically you're

Attacks – RSA Timing Fixes

- Fix
  - Only use multiply (A*A = A^2)
  - Dummy multiplies
- Possible Issues
  - A*A has a different power profile than A*B
  - Compiler may remove the dummy operation

2/27/2016 Don Matthews

has a different power profile than A times B, and I had to

Attacks – Discussion Points

- Power/EM analysis
  - Don't need all the bits can use analysis plus brute force
- Cache Attacks
  - Some implementations use tables

2/27/2016 Don Matthews

99% of it's already there for you, okay?

Threat Model – Page 2

- What are the attack avenues
  - Who possesses it
    - Owner or User
  - Access
    - Fixed location or mobile
    - Visible location (people might observe someone tampering with it)
  - Internet enabled
  - Wireless

12/12/2016 Don Matthews

Stored value card,
it's in your possession, okay.

Attacks Again

- Let's go into some additional attacks
- Remember Dave's Electronic lock?
- What can go wrong with it?
- How do I go from 10K combinations to 24?

12/12/2016 Don Matthews

Besides the,
I can potentially hack the bluetooth.

His guess was the code was 1968,

More Attack Discussion Points

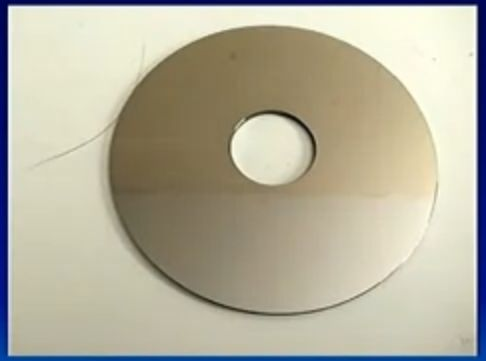
- RFID chip (DefCon)
  - Attackers have time on their hands
- Key Fob
  - Man in the middle attack
- EEPROM
  - Reset the security bit
- USB encryption device
  - Do I need to know your key

12/12/2016 Don Matthews

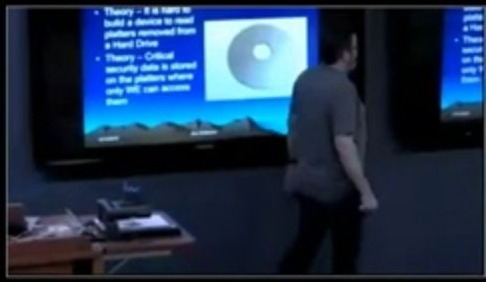
Now, I told you about the DefCon chip. How many have cars?

Hard Drives

- Theory – It is hard to build a device to read platters removed from a Hard Drive
- Theory – Critical security data is stored on the platters where only WE can access them



12/12/2016 Don Matthews



is hard to build a device to read platters removed from a hard drive.

Hard Drives – Pg 2

- Data Recovery companies build them and will recover data for you – Thousands of dollars
- Fortunately we don't need to do that, the drive companies have given us a tool to do that – The Hard Drive

12/12/2016 Don Matthews

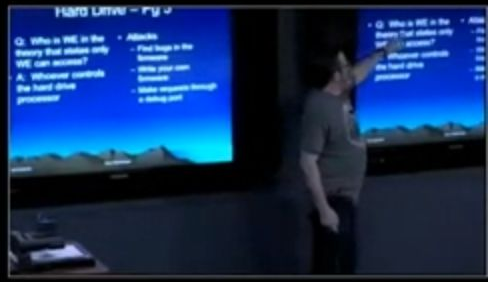
I can read anything I want on there, can I?

Hard Drive – Pg 3

- Q: Who is WE in the theory that states only WE can access?
- A: Whoever controls the hard drive processor

- Attacks
  - Find bugs in the firmware
  - Write your own firmware
  - Make requests through a debug port

12/12/2016 Don Matthews



who is "We" in the theory that states only "We" can access it?

Attacks Again

- Let's go into some additional attacks
- Remember Dave's Electronic lock?
- What can go wrong with it?
- How do I go from 10K combinations to 24?

12/12/2016 Don Matthews

Okay. Well, most people aren't fine tuned.

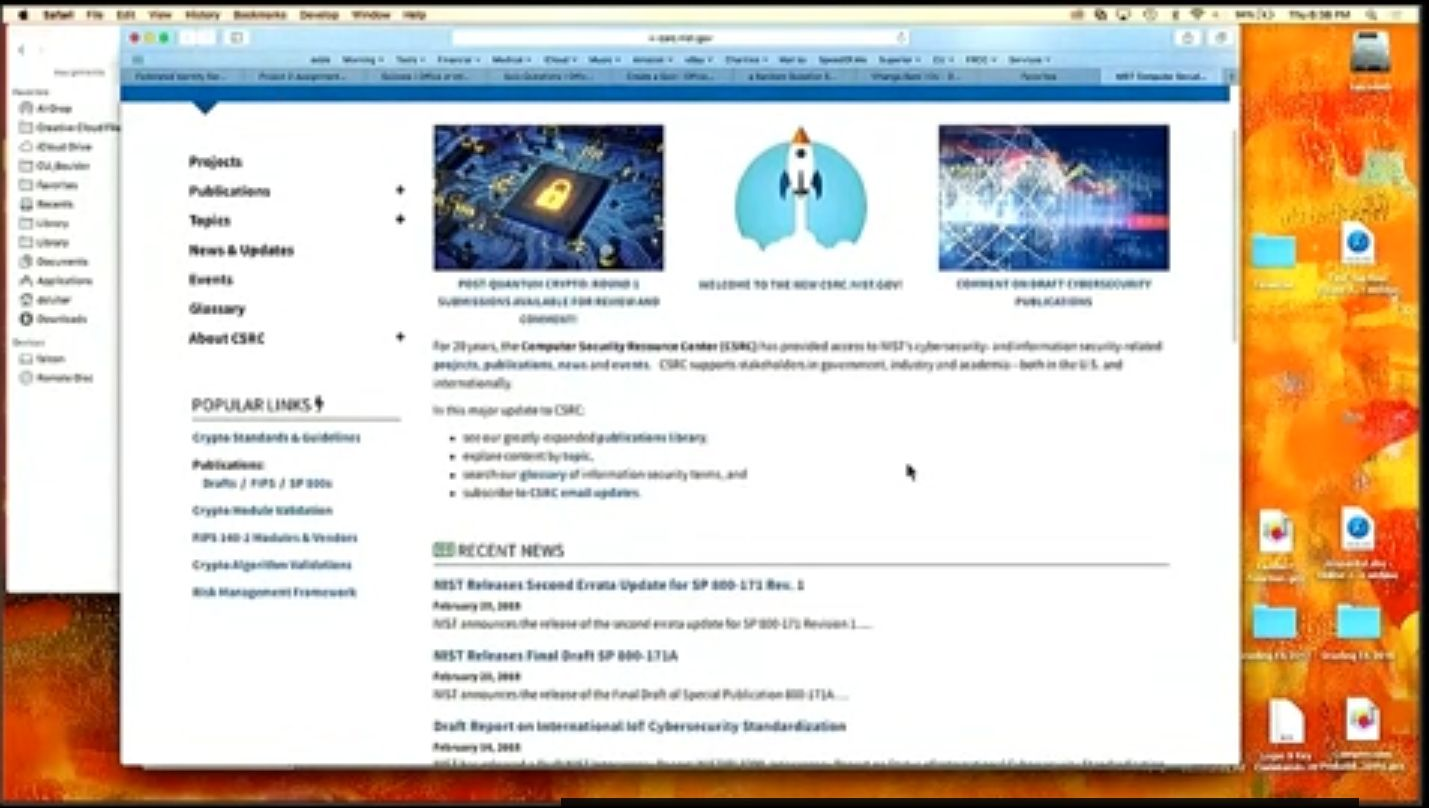
Password Table Attacks

- Dave discussed the use of hashing algorithms for password checking
- Can you spot a problem with this password table?

USERNAME	HASHED PW
JOE	DYECAEYFN
LUCY	JZEDHVUE6
XI	HEIVC83ND
AMIT	C8DNADEVY
ANU	DYECAEYFN

12/12/2016 Don Matthews

Unless, I get one of those little cameras and I put it



There's a ton of information here.

Quick Links

- Draft Pubs
- Final Pubs
- FIPS
- SPs (Special Pubs)
- NISTIRs
- ITL Bulletins
- White Papers
- Journal Articles
- Conference Papers
- Books

Search

Search publication record data
(not a full text search)

Sort By

Number (highest to lowest)

Results View

Brief

Search Results

Status: ✖ Draft ✖ Final Series: ✖ FIPS
Showing 9 matching records.

Series	Number	Title	Status	Release Date	Relevance
FIPS	202	**SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions** Download: FIPS 202 (DOI); Local Download; Federal Register Notice; Press Release; Comments received on Draft FIPS 202; Draft FIPS 202 (May 2014)	Final	8/04/2015	100%
FIPS	201-2	**Personal Identity Verification (PIV) of Federal Employees and Contractors** Download: FIPS 201-2 (DOI); Local Download; 2012 Draft Comments and Dispositions; 2011 Draft Comments and Dispositions; Revised Draft (July 2012); Draft FIPS 201-2 (March 2011)	Final	9/05/2013	100%
FIPS	200	**Minimum Security Requirements for Federal Information and Information Systems** Download: FIPS 200 (DOI); Local Download	Final	3/01/2006	100%
FIPS	199	**Standards for Security Categorization of Federal Information and Information Systems** Download: FIPS 199 (DOI); Local Download	Final	2/01/2004	100%

US Security Specifications

- NIST (National Institute of Standards and Technology), see: https://www.nist.gov
- FIPS (Federal Information Processing Standards) and SP800s (Special Publications), see Computer Security Resource Center: https://csrc.nist.gov
- A recent addition is SP800-193, Platform Firmware Resiliency Guidelines, protect against:
  - Unauthorized changes
  - Detecting unauthorized changes
  - Recovery from attacks
- Microsoft has an initiative as well: https://www.microsoft.com/en-us/research/publication/cyber-resilient-platforms-overview/

There's a link to their cyber resilient platforms overview.

Summary

- Develop a Security Mindset, apply "orthogonal" thinking
- Be clear about what are you trying to protect
  - Information/Communication
    - Use encryption
  - Authenticity / Authentication
    - Use MACs
  - Integrity
    - Use Hashes
- Use the current algorithms that are thought to be "secure" at the time
- Address security at all levels and all interfaces in a system
- Always authenticate software/firmware updates - build this into your systems from day one
- Key management is the hard part, but also where you can be creative
- Security is only ever "good enough"
- Security is a never-ending game of cat and mouse
- For more in-depth learning take Eric Wustrow's course "*Introduction to Computer Security*"

Introduction to Computer Security. He's really good.