Protecting the Client

CS@CU

Protecting the Client

- Clients of networked applications

- Smart cards

- DRM

Network Clients

- Thus far, we've focused on servers—network apps and privileged programs

- Do clients have the same risks?

- Yes—in some ways, more. . .

Clients at Less Risk

- You can send arbitrary text to a web server

- To attack a web browser, you have to lure the user to the infected server

- But—email can be sent to anyone—and it can contain links to web servers

"Watering Hole" Attacks

- Hack a web site that you think your targets will visit

- Or—buy an ad through an ad broker; have this ad serve up malware

☞ This has happend to major news sites, including the *New York Times* and *Forbes*

- Wait...

Email and the Web

- Most mailers do not do their own HTML processing (at least not for full HTML)

- They have some way of invoking the standard browser's rendering engine, and hence are at risk from browser security flaws

- Example: the Eudora mailer suffered from an Internet Explorer flaw (`http://email.about.com/cs/eudoratips/qt/et122001.htm`)

- Mozilla provides a rendering engine for use by other applications

- iOS has one for use by apps

- Active content—Java, Javascript, ActiveX, Flash, and the like—make life *much* harder (but we'll cover that later in the semester)

Other Applications

- *Any* program, including network clients, can have security flaws

- Buffer overflows and the like abound

- If someone using the application connects to your server—or if you can trick someone into connecting to your server via a *watering hole attack*—you can compromise their machine

- Client software is often updated *less* frequently

- Rarely run on dedicated machines

The Enemy Has Your Device

- Normally, you own your machine

- What if the enemy owns it?

- What if your enemy is *supposed* to own it?

- Who'd do that? Lots of people...

Smart Cards

- Often in credit card form factor

- Contains small CPU and non-volatile storage (some contain an RSA accelerator chip)

- Used for many purposes

- Note: not the same as RFID-based cards

Reader and Smart Cards
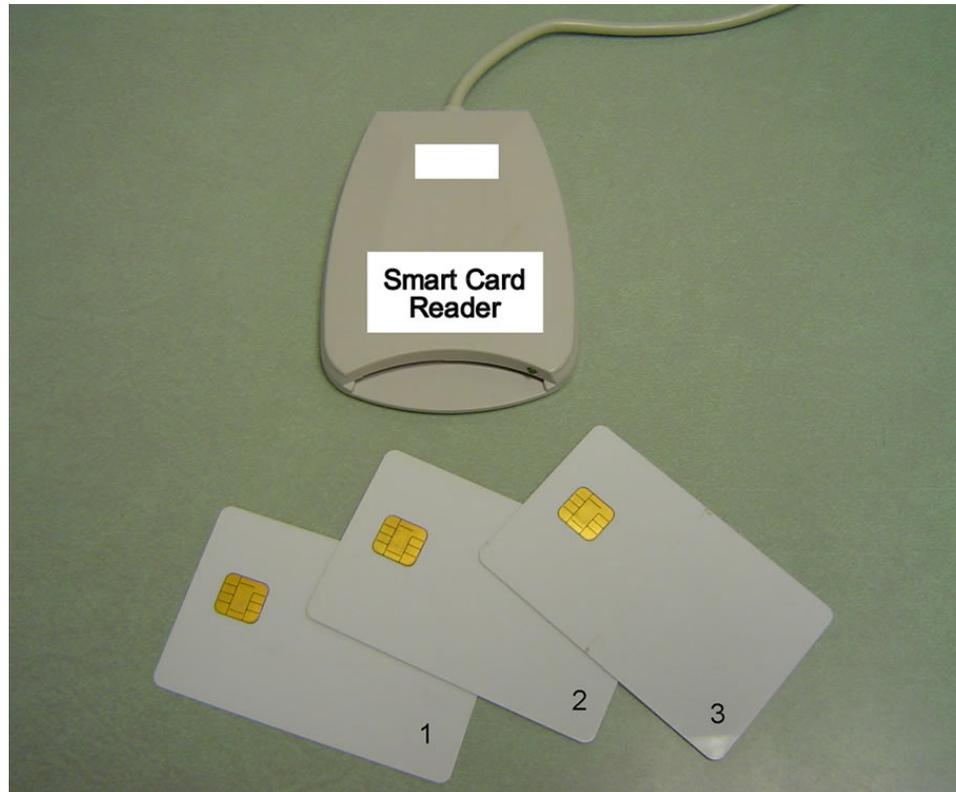
These cards use contacts for power and I/O. Other types use inductive coupling or radio.

Uses of Smart Cards

- Money or equivalent (transit fares, parking meters, vending machines, etc.)

- Resistant to counterfeiting (credit cards, especially in Europe and now being deployed in the US)

- "Something you have" for authentication

- Pay TV or satellite/cable box

The Enemy's Goals

- In stored value systems, the attacker wants to add more money to the card

- Alternatively, the attacker wants to extract the secret, to permit counterfeiting of more cards

- If an authentication token is locked by a PIN, the attacker wants to discover or replace the PIN

Attack Techniques

- Software

- Physical

Software Attacks on Smart Cards

- The card is running software and talking to the outside world

- Can you launch a buffer overflow attack?

- Is that software vulnerable to the usual attacks?

- Note: most smart cards don't have a privileged mode and a protected kernel. (What about multi-use smart cards?)

- In principle, any attack that works against other systems can work here

Physical Attacks

- An attacker can manipulate the physical environment

- An attacker can monitor the physical behavior

- An attacker can try to reverse-engineer the chip

Differential Power Analysis

- When a transistor switches on, it allows current flow

- The power difference between "on" and "off" can be measured

- Measure the instanteous power consumption of the smart card

- Repeat this for a moderately large number of operations

- Statistical analysis will reveal the value of the cryptographic key at different points

Fault Injection

- Certain stresses can cause erroneous computations

- It's practical to use heat or radiation to confuse a CPU or memory

- It has been demonstrated, theoretically and practically, that this can be used to break security

- Other techniques: controlling power

Reverse Engineering

- With suitable tools, it's possible to reverse-engineer a CPU and read the memory

- Techniques include microtomes, scanning electron microscopes, and the like

- Separate set of physical-layer defenses

From Chipworks.com's Web Site

"Chipworks' Circuit Analysis Reports provide full reverse engineering information of all or selected circuitry on a particular die—complete with detailed schematics to reveal the operation of a competitor's device. . .

"Report Contents:

- "Package and die overview including a package x-ray.

- "Annotated die photograph identifying the major functional blocks.

- "Complete set of hierarchical schematics to capture the operation of the device.

- "Top-level schematic overview to capture the entire function of the circuit.

- "Schematics for each analyzed block."

Protecting Chips

- Use of "non-metallic links" for device programming

- Program in sensitive data; don't hardwire it.

- Ghost logic—fake logic elements to complicate the analysis

- False heat dissipation

- Extra metal layers

Satellite and Premium Cable TV

- *Very* sophisticated enemies

- General attack

    - Sell counterfeit descrambler boxes

    - Buy legitimate access card

    - Extract key from card

    - Distribute keys over the Internet

Where is the Value?

- Is the value on the card or in some database?

- If the value is in the card, must take precautions against counterfeiting

- If the value is in the database, the card is just a pointer; security features in the card protect against theft, not value loss

MTA Metrocard

Primarily an online system; central database has authoritative card value. Some offline use for buses. No protection against theft for pay-as-you-go Metrocards; database update for monthly cards.

Washington, D.C., Farecard

Note that the card itself knows its value. No protection against theft or counterfeiting.

The CU ID Card

- The RFID chip and mag stripe are primarily database indices

- (Older cards had a bar code, too, but the mag stripe held more data. Both had social security numbers.)

- If the card is reissued, a new database index is assigned

Mag Stripe Cards

- Two or three tracks; standardized format

- Old CUID had social security number plus other fields:

```
track 1: error: e5
track 2: <2118713710312940>
track 3: error: e5
```

- My Amtrak card:

```
track 1: <AGR^STEVEN^BELLOVIN^PLUS>
track 2: <20170228=5081658710>
track 3: error: e5
```

(Note the expiration date on track 2)

Mass Transit Fare Cards Broken

- Dutch researchers cracked their public transit system's RFID cards, both cryptographic and non-cryptographic

- MIT students cracked the MBTA's (Boston) systems: mag stripe and cryptographic RFID

- Many other transit systems vulnerable

- Why? Bad crypto, underestimating the enemy, simple mistakes of the type we've discussed here

RFID Chips in Passports

- New passports contain an RFID chip with data and digitized picture (ICAO standard)

- This data is, of course, digitally signed

- But—not all countries have signing certificates for all other countries

- Permits the creation of passports with forged RFID chips

- Solution: creation of a PKI, where each certificate lists the countries for which it can sign passports. But—is this PKI trustable?

- Who is the enemy? Smugglers? Terrorists? Foreign intelligence agencies? All of the above?

Apple iPhone Encryption

- All content is encrypted with one of a set of randomly-generated AES keys

- These keys are themselves protected: either encrypted with a key derived from a random UID that is stored in a secure, on-chip area, or encrypted with a key derived from the UID and the PIN

- Hardware-enforced maximum guess rate of 80 ms/try

The FBI versus the iOS 8

- Assume custom software, either provided by Apple or via an FBI jail-break of the phone

- Try all possible PINs:

4 digits	800 seconds
6 digits	22 hours
6 lower-case letters or digits	5.5 years
6 letters or digits	144 years
8 arbitrary characters	253,678 centuries

☞ These numbers assume *random* PINs. Is that assumption valid? Almost certainly not:

`https://freedom-to-tinker.com/blog/jbonneau/`
`guessing-passwords-with-apples-full-device-encryption/`

Digital Rights Management

- Allow publisher to control use of content

- Prevent arbitrary redistribution of copyrighted materials

- Change sales terms from *physical purchase* to *license*

General Approaches

- Restrict consumer's ability to use the material

- Trace usage (often via "watermarking")

Restricting Use

- Preferred approach of content providers

- Used by most (legitimate) vendors of digital films and books; formerly used by music vendors

- Many different types, implementing many different policies

Apple's iTunes

- "personal, noncommercial use"

- "five Apple-authorized devices at any time"

- "shall not be entitled to burn video Products or ring tone Products"

- "burn an audio playlist up to seven times"

- (`http:`
  `//www.apple.com/support/itunes/legal/terms.html`)

Microsoft's Media DRM

- Media files are encrypted, and contain pointer to license source

- User obtains license from clearing house

- License includes terms and conditions as well as decryption key

- "Licenses can have different rights, such as start times and dates, duration, and counted operations."

- "may allow the consumer to . . . copy the file to a portable device"

- "Licenses, however, are not transferable."

- (`http://goo.gl/CEMLu`)

How Does It Work?

- Operating system mediates access to files

- Operating system enforces rules imposed by the content provider

- Ordinary OS protection mechanisms isolate the unprotected content from the user

- Or do they?

The User Versus the OS

- If you own the computer, you're the administrator; you have root privileges

- The vendors' challenge: protecting content against the superuser

- Several different approaches

Approaches

- Obscurity—make it hard to find the plaintext

- Obfuscation—confuse the code to make reverse-engineering harder

- End-to-end crypto—do decryption on the sound card or video card. (But what abou the "analog hole"?)

- Trusted hardware

- Automatic upgrades

Automatic Upgrades

- Many carriers and vendors do that to mobile phones

- Microsoft and Apple will sometimes update your software without telling you

- DVR vendor took away "skip 30 seconds" button

- Upgrades or downgrades?

Trusted Hardware

- Ultimately, all software-only schemes are futile—but is it *good enough*?

- You can always trace the code, patch modules, etc

- But—other software can attempt to detect such "attacks", and disable playback

- See above

- The *only* reliable solution is trusted hardware

- Manage the keys and the decryption outside of the OS

- As needed, use tamper-resistance techniques for such hardware

Watermarking

- Tag files with owner and/or licensee information

- Tags should should be invisible in normal use of the file

- Tags should resist detection and deletion

- Used for iTunes' unlocked music files

- In practice, this has proved to be extremely difficult to accomplish

Defeating Watermarking

- Pictures: scaling, clipping, color balance, rotation, geometric distortion, printing/scanning

- Sound: Fourier transforms

- Thus far, the attackers are winning in theory, but many consumers buy watermarked music

Economics vs. Algorithms

- "Amateurs worry about algorithms; pros worry about economics"

- What does it *cost* the attacker to break the DRM or watermarking?

- If the cost is high enough and the price for legitimate purchase is low enough, most people won't bother to break the DRM

- Is that good enough, given how rapidly files can spread? Apple thinks so, and has persuaded content owners of that

Legal Issues

- The law in the US and other countries outlaws "circumvention technology"

- Lawsuits and threats of lawsuits have blocked some work

- A lot of other stuff is out there, including both academic research and practical tools (i.e., dcss)

- Crucial philosophical issue: do DRM schemes give content owners more power than copyright law would?

Current State of Affairs

- Technical measures are good enough that they're not the weak point

- Security need not be perfect, merely good enough

- CD/DVD ripping plus redistribution is easier than cracking DRM schemes

- That said, many DRM schemes have been cracked

- Two-fold attack by content owners: technical measures and lawsuits

Conclusions

- Many situations require that some crucial device be in enemy hands

- True security in such cases is almost impossible

- More than in most situations, this shows the role of economics: the attack has to be cost-effective

- This is usually the best approach: make the security *good enough*