

Insecurity at U of M
HKN Tech Talk
Jon Oberheide
CTO, Duo Security

Who am I?

Jon Oberheide

- ▶ BS, MS, and PhD from U of M
- ▶ Broken a bunch of stuff here and elsewhere

Duo Security

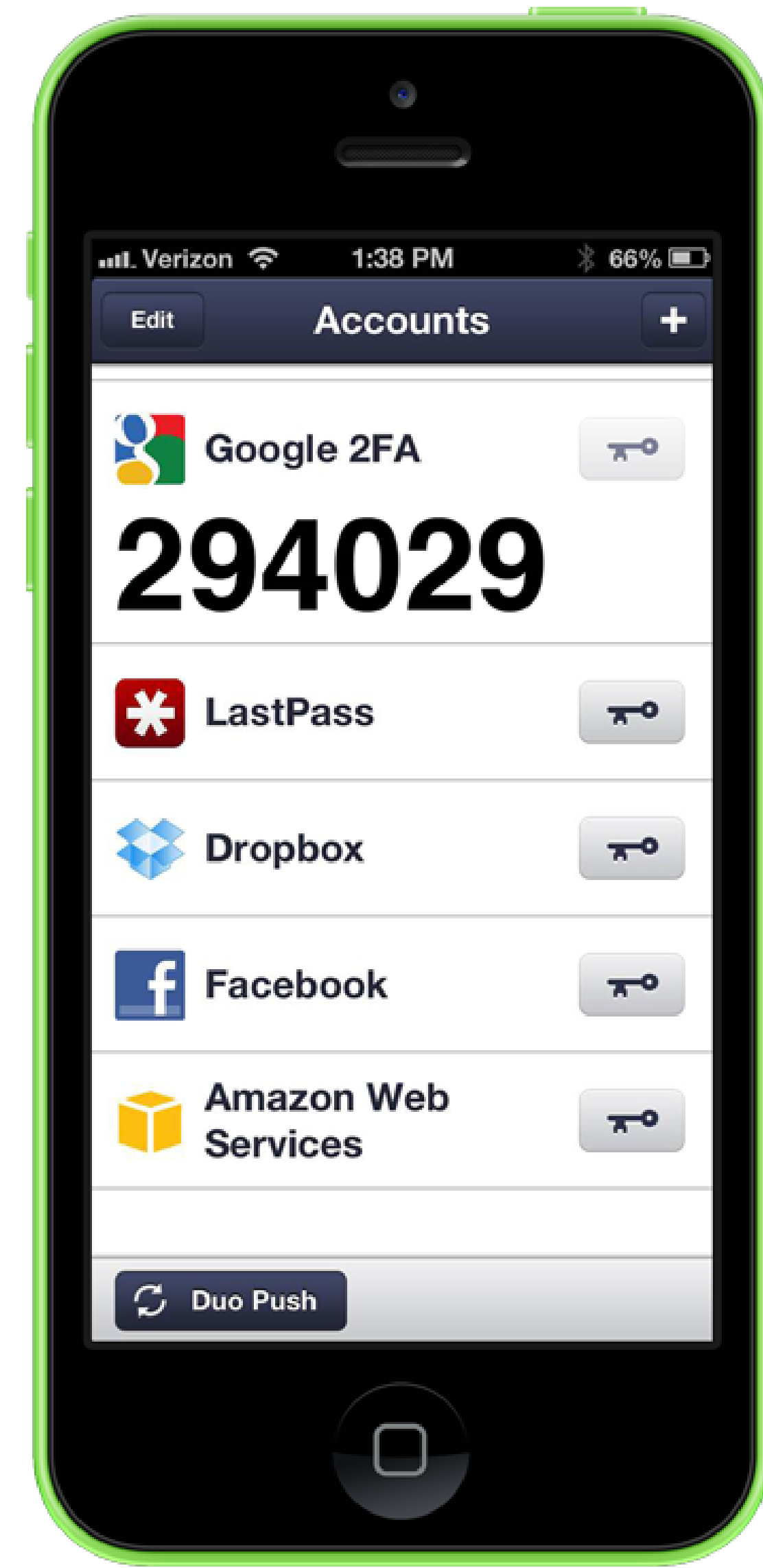
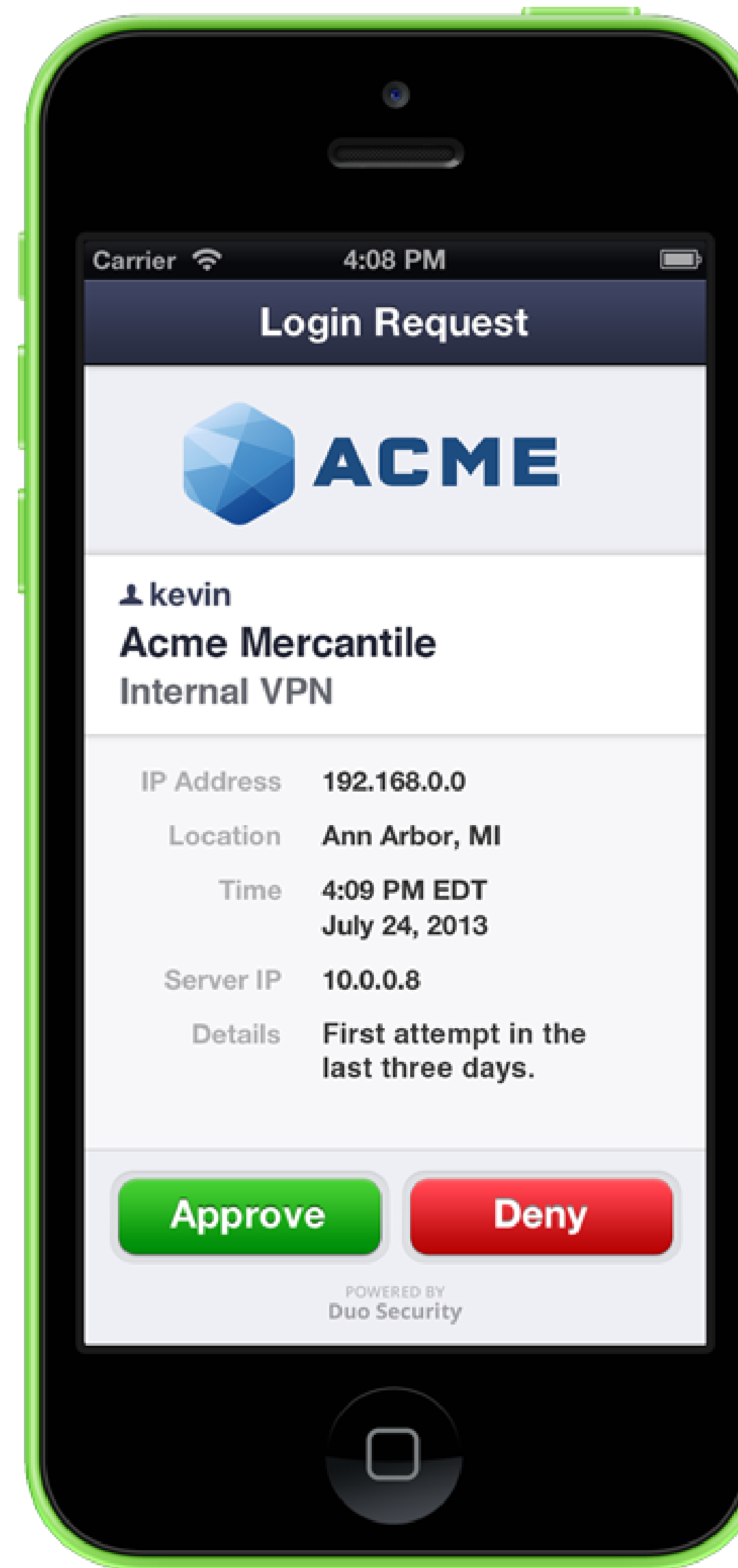
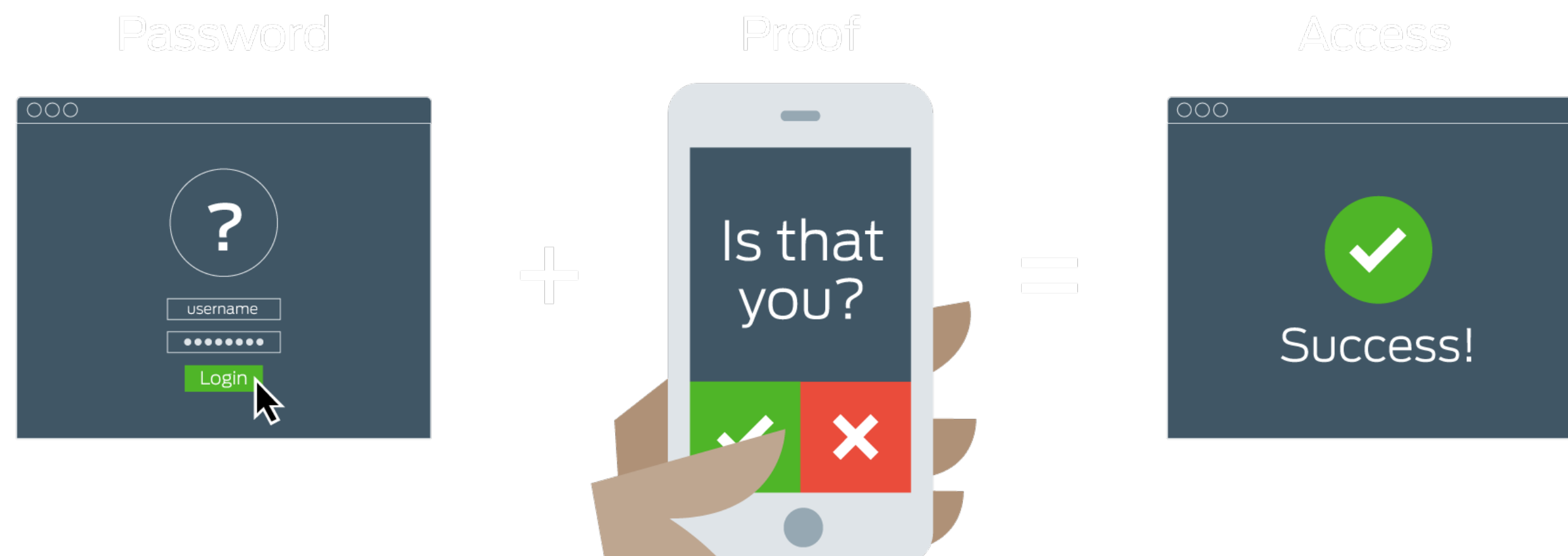
- ▶ Ann Arbor-based, founded in 2009, growing rapidly!
- ▶ Technology, customers, culture!
- ▶ RECRUITING, RECRUITING, RECRUITING!



What is Duo?

Two-factor authentication

- ▶ What you know + what you have
- ▶ Stops NSA, China, hoodlums



Why should we care about security?



UNIVERSITY OF MICHIGAN
WOLVERINE ACCESS

[Home](#) [Help](#)

Find an Existing Value

EmplID:

Last Name:

First Name:

National ID:

[Basic Search](#)

[Add a New Value](#)

Search Results

View All First 1-2 of 2 Last

EmplID	Name	National ID
[redacted]	Oberheide, Kristin Elizabeth	[redacted]
6891	Oberheide, Jonathan Clarke	[redacted]



Security at the University

EDUs are soft targets

- ▶ With lots of valuable personal information
- ▶ Including your personal information
- ▶ In your best interest to ensure (responsibly!) that this information is securely protected

Let's explore a few vulnerabilities at UM

- ▶ Doesn't take l33t skillz
- ▶ Just a bit of curiosity and free time



Agenda

Cosign authentication bypass

Mcard forgery attack

Physical security of CSE

Other things...



Cosign SSO Vuln



Cosign Single Sign-On Authentication

- ▶ Deployed extensively at UofM and many other educational institutions and orgs around the world
- ▶ Protects web mail, wolverine access, mfile, mpathways, and umm...everything



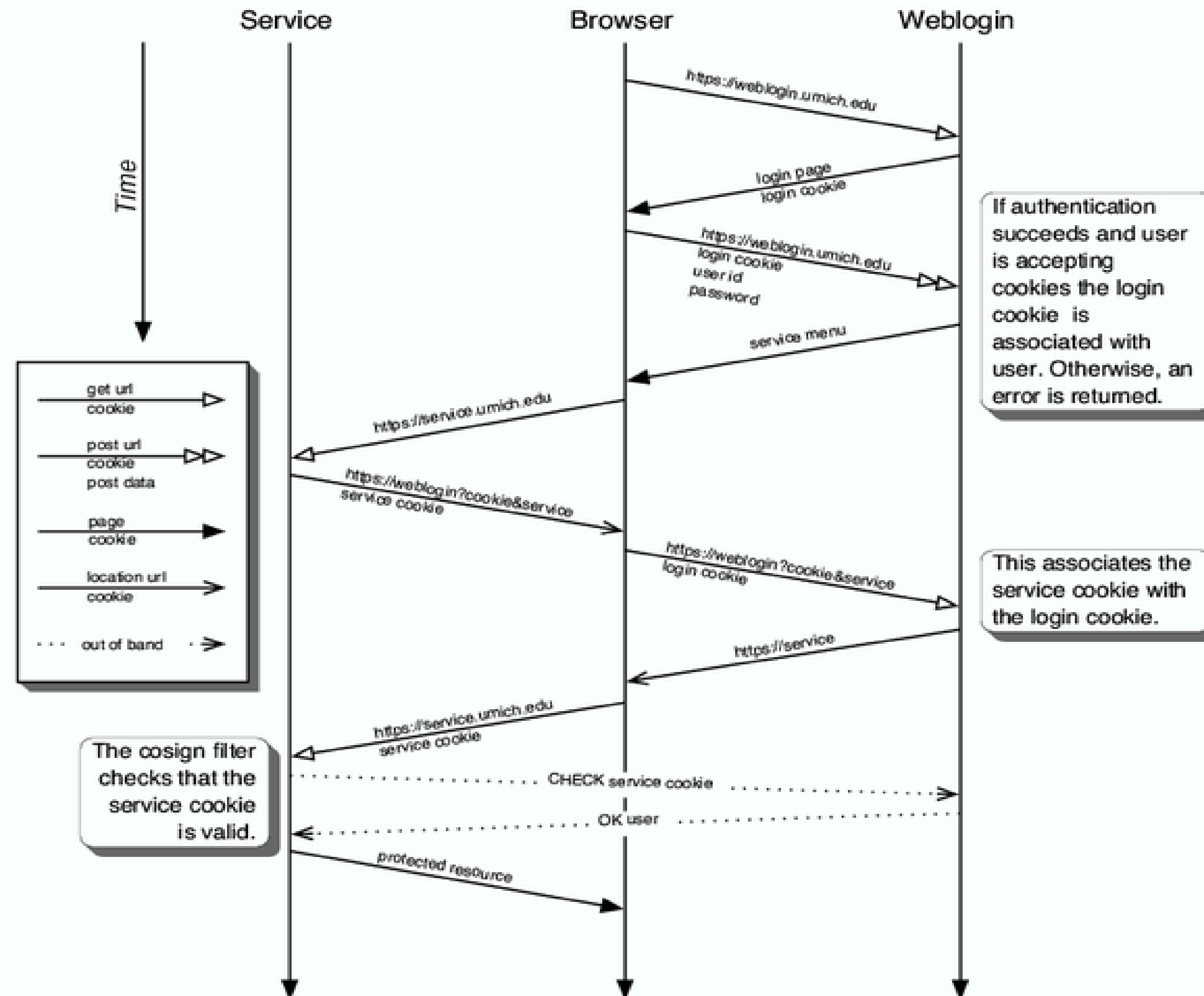
This is Cosign

The screenshot shows a Mozilla Firefox browser window titled "weblogin: - Mozilla Firefox". The address bar displays "https://weblogin.umich.edu/". The browser's menu bar includes File, Edit, View, History, Bookmarks, Tools, and Help. The toolbar contains navigation buttons (back, forward, home, stop, refresh) and various utility icons like Cookies, CSS, Forms, Images, Information, Miscellaneous, Outline, and Resi. The main content area features the University of Michigan logo and the text "UNIVERSITY OF MICHIGAN WEBLOGIN". Below this, a message states "AUTHENTICATION REQUIRED::" and explains that a U-M website requires authentication. It asks the user to enter their Login ID (username or Friend ID) and password. A section titled "Need a Login ID?" provides a link to "create one now". To the right, a light blue box contains a login form with fields for "Login ID", "Password", and "MToken", a "Log In" button, and links for "Forgot your password?" and "Login Help". At the bottom of the page, a dark blue footer contains the text: "U-M Gateway | Copyright © 2007 The Regents of the University of Michigan, Ann Arbor, MI 48109 USA 734-764-1817", "University of Michigan-Dearborn, MI 48128 USA 313-593-5000", and "University of Michigan-Flint, MI 48502 USA 810-762-3000". The browser's status bar at the bottom shows "Done" on the left and "weblogin.umich.edu" on the right.



Cosign Architecture

Case 1: User Visits Weblogin First



Cosign Architecture

Cosign-enabled Webservers (eg. web mail)

CGI Web Frontend (weblogin.umich.edu)

Backend Daemon (cosignd)

Hugely simplified operation:

- ▶ Webserver redirects user to CGI web frontend for auth
- ▶ CGI authenticates user, then communicates with daemon
- ▶ Webserver checks with daemon to verify authentication



Cosign Protocol

Protocol between CGI and daemon:

- ▶ Plaintext-based protocol, SMTP-ish
- ▶ Commands terminated '\n', '\r', or '\r\n'

Example commands:

- ▶ CHECK: check whether a given cookie is valid in daemon's backend db (eg, if a user is already auth'ed)
- ▶ LOGIN: tell daemon a user has auth'ed
- ▶ REGISTER: associate service cookie with user's global cosign cookie



Typical Login Session

Exchange between CGI and daemon:

- ▶ CHECK cosign=X
- ▶ LOGIN cosign=X 1.2.3.4 username
- ▶ REGISTER cosign=X 1.2.3.4 cosign-service=Y

where X and Y are randomly generated base64 strings of length 128, username is the principal that successfully authenticated, and 1.2.3.4 is the IP associated with the user



Cosign Audit

Had some free time one weekend

C-based CGI component

- ▶ It's C-based parsing of HTTP, it has to be buggy
- ▶ Yet, surprisingly well-coded

String-based communications protocol

- ▶ Typically not as fruitful as binary proto, but eh
- ▶ Maybe some unsafe string handling in daemon



Audit Findings

Lots of NULL derefs triggerable in CGI

- ▶ No big deal, CGI spawned off on each request

Buffer overflow in daemon

- ▶ Unfortunately, not enough for stack smashing, only can overflow into subsequent buffer on stack
- ▶ Overflows into krb_ticket var which is unlink()'ed
- ▶ May be exploitable with other archs/stack layouts

No exploitable common C-based vulns,
anything application specific???



Initial CHECK Command

When you hit `weblogin.umich.edu`

- ▶ CGI takes your HTTP cookie and sends to daemon
- ▶ `send_daemon("CHECK %s\r\n", cookie);`
- ▶ Needs to check whether presented cookie is valid

Remember our protocol line terminators???

- ▶ `'\n'`, `'\r'`, `'\r\n'`

Your security-sense should be tingling now...



Embedded Terminators

Can't embed '\n' in HTTP header fields (cookie)

But carriage returns '\r' are completely legal

- ▶ Set HTTP cookie
 - ▶ `cosign=blah\rred\rgreen\rblue`
- ▶ CGI sends to daemon:
 - ▶ `CHECK cosign=blah\rred\rgreen\rblue\r\n`
- ▶ Daemon interprets as four separate commands due to '\r'
 - ▶ "CHECK cosign=blah", "red", "green", "blue"



Uh-oh...

Just achieved arbitrary Cosign command execution!

- ▶ From a completely unauthenticated web user

How to exploit?

- ▶ Need to replicate the standard login procedure
- ▶ Inject LOGIN, REGISTER command sequence



Exploitation

Example malicious cookie:

- ▶ `cosign=X\r`
LOGIN `cosign=X 1.2.3.4 username\r`
REGISTER `cosign=X 1.2.3.4 cosign-servicename=Y`
- ▶ Replace username with the uniqname you want to authenticate as ('marysuec', 'grue', ...)

Success!

- ▶ Steal personal information, change your grades, read email (and that's the tame stuff...)
- ▶ MPathways is where the fun is



Agenda

Cosign authentication bypass

Mcard forgery attack

Physical security of CSE

Other things...



Magnetic Card Security

Magnetic Cards

- ▶ Trivial to clone given physical access

House key analogy

- ▶ Copies made at hardware stores
- ▶ If attacker obtains your physical key, he can make a copy and break into your house
- ▶ Obvious!



Mcard Hacking

So, can we forge a Mcard without a physical copy/clone of it?



First, lets take a look at what the Mcard magnetic stripe contains...



Mcard Format

Magnetic-Stripe Card Explorer

File Setup About

Scan Port for Data Autoexit
Stop Scanning Error correction

Swipe Characteristics
Swipe time: 164 ms
Swipe speed: 51 cm/s
44 <speed> 58 cm/s
100
#1
#2
#3
10

Status: Ready

Decode Char | Signal Analysis | Data Analysis | Write Track

Track#1 213 BPI %B6008476891430820^OBERHEIDE/J^1106120?Q
Char set: ALFA B6008476891430820
Chars: 40 OBERHEIDE/J
Parity: Ok 1106120
LRC: Ok

Start Sentinel : %
Data : B6008476891430820
Field Separator: ^
Data : OBERHEIDE/J
Field Separator: ^

Track#2 75 BPI .6008476891430820=1106120=091564875?2
Char set: BCD 6008476891430820
Chars: 37 1106120
Parity: Ok 091564875
LRC: Ok

Start Sentinel : ;
Data : 6008476891430820
Field Separator: =
Data : 1106120
Field Separator: =

Track#3
Char set:
Chars:
Parity:
LRC:



Mcard Format

Magnetic-Stripe Card Explorer
File Setup About

Scan Port for Data **Autoexit**
Stop Scanning **Error correction**

Swipe Characteristics
Swipe time: 164 ms 100
Swipe speed: 51 cm/s #1
44 <speed> 58 cm/s #2
10 #3

Status: Ready

Decode **Signal Analysis** **Data Analysis** **Write Track**

Position < | > Zoom | >

; 6 0 0 8 4 7 6 8 9 1 4 3 0 8 2 0 = 1 1 0 6 1 2 0 = 0 9 1 5 6 4 8 7 5 ? 2

Select Track
 Track#1
 Track#2
 Track#3

Display options
 Auto - Zoom

Track density (BPI): 75
Total number of Ticks: 323
First "1" Bit found at position: 24
Character Set found: BCD

Tick	Char	Nr.	Flux	us	Bit
1		0	0	4097	0
2		0	1	716	0
3		0	0	813	0
4		0	1	761	0
5		0	0	813	0
6		0	1	776	0
7		0	0	784	0
8		0	1	761	0
9		0	0	776	0
10		0	1	761	0
11		0	0	776	0
12		0	1	716	0
13		0	0	746	0

Tick Nr. 0/0 Tick duration



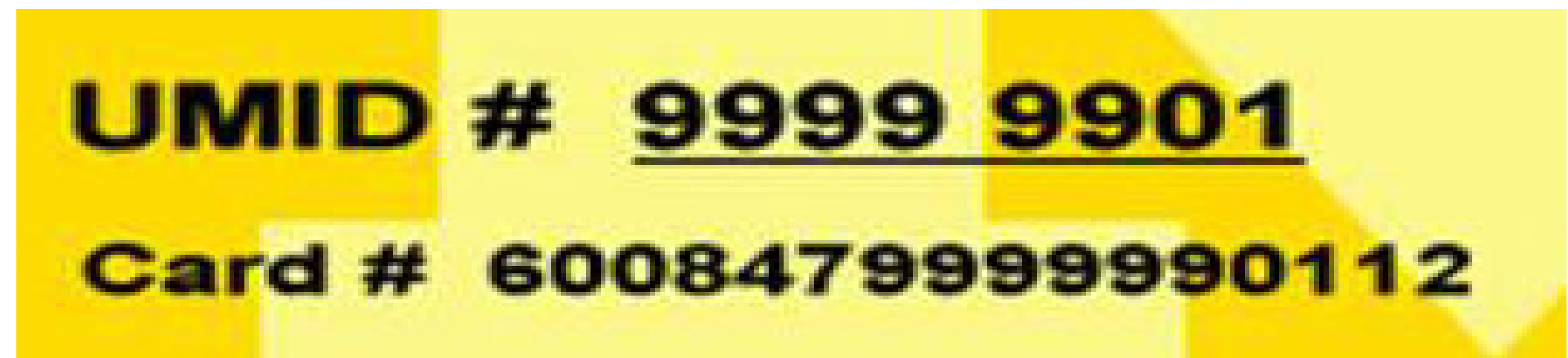
Mcard Format

What do the Mcard readers care about?

- ▶ Only data from track 2 is read
- ▶ Only the account number portion is verified
- ▶ Lots of trial and error...and candy!

Mcard account number

- ▶ 16 digits, listed on front of card
- ▶ Static 6-digit prefix: 600847
- ▶ Then 8-digit UMID: 99999901
- ▶ Then card revision number digit
- ▶ Finally, Luhn checksum digit



Mcard Forgery Attack

Implications?

**The account number is completely predictable.
We can forge arbitrary Mcards!**

All we need:

- ▶ Victim's UMID
- ▶ Public info, lookup via web or uns service
- ▶ Revision number
 - ▶ Usually 1 or 2, worst case ten guesses
- ▶ Luhn checksum
 - ▶ Standard algorithm, trivially calculated from other 15 digits



Find Target's Uniqname

University of Michigan Online Directory - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://directory.umich.edu/ Google

Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize

UNIVERSITY OF MICHIGAN ONLINE DIRECTORY

Jon Oberheide Search

[more choices](#)

Bind Add Delete Modify Help


not bound: click 'bind' to make change

The University of Michigan Online Directory (UMOD) is a database of faculty, staff, students, alumni and groups. Use the search box above to search by e-mail address, telephone number, full name (in firstname lastname order), surname, or uniqname.

News & Tips

- [Directory to display preferred names \(4/25/07\)](#)
- [New policy: Groups must be renewed yearly \(02/19/07\)](#)

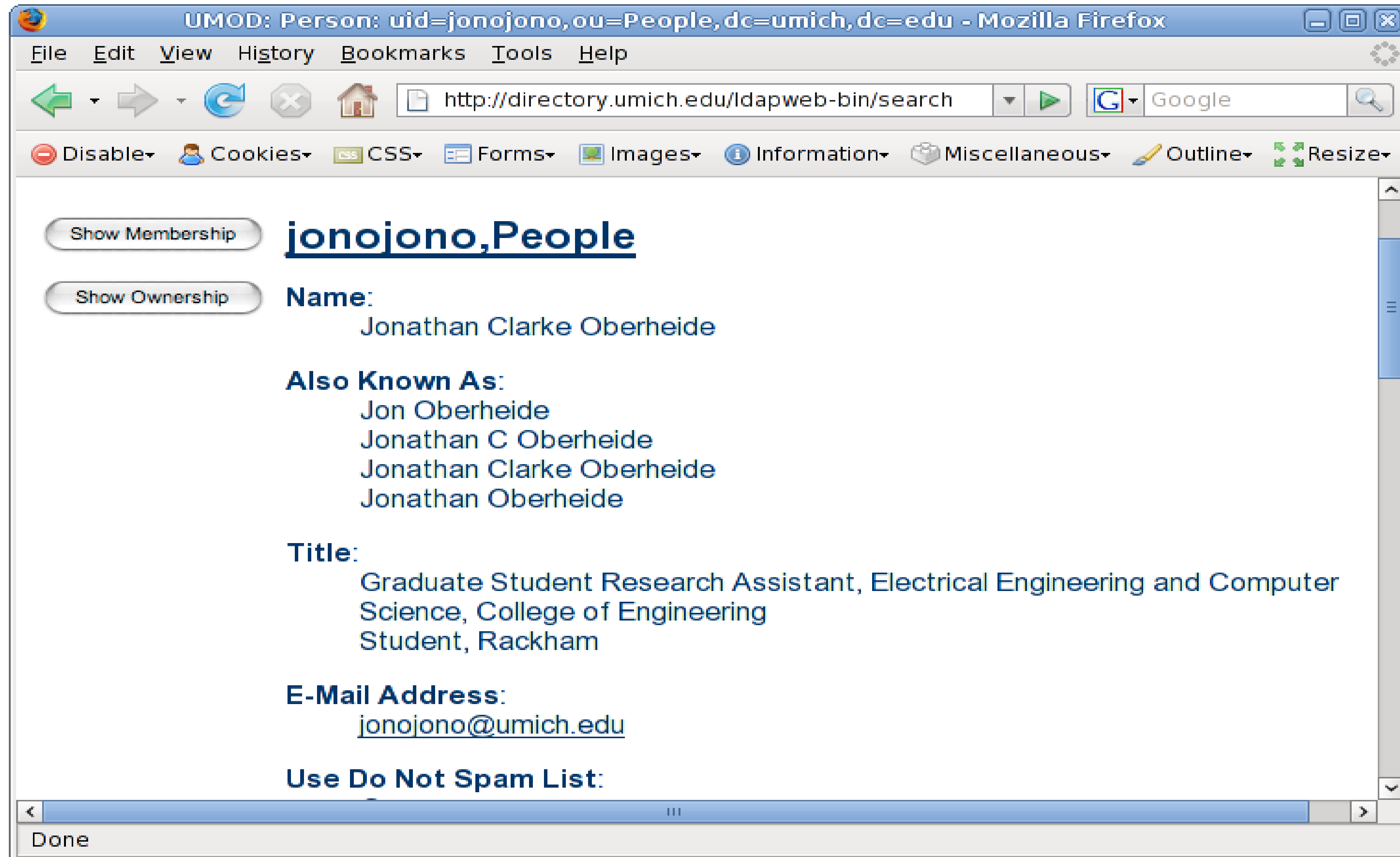
Hint: Bind



Done



Find Target's Uniqname



The screenshot shows a Mozilla Firefox browser window with the following details:

- Address Bar:** `http://directory.umich.edu/ldapweb-bin/search`
- Page Title:** UMOD: Person: uid=jonojono,ou=People,dc=umich,dc=edu
- Buttons:** Show Membership, Show Ownership
- Uniqname:** **jonojono,People**
- Name:** Jonathan Clarke Oberheide
- Also Known As:** Jon Oberheide, Jonathan C Oberheide, Jonathan Clarke Oberheide, Jonathan Oberheide
- Title:** Graduate Student Research Assistant, Electrical Engineering and Computer Science, College of Engineering Student, Rackham
- E-Mail Address:** jonojono@umich.edu
- Use Do Not Spam List:** (checkbox)



Find Target's UMID

```
jonojono@dionysus:~  
File Edit View Terminal Tabs Help  
* The Login and SFTP servers are rebooted weekly at 4:00 A.M. on Sundays. *  
*****  
*****  
* Were you logged in without being prompted for your password? Your client *  
* is probably configured to use your kerberos credentials for authentication. *  
* If this occurred and you want to change the behavior, or if you are *  
* interested in more details, please contact itd.login.admins@umich.edu *  
*****  
Welcome to Linux!  
bash-2.05b$ uname -a  
Linux zaxxon.gpcc.itd.umich.edu 2.6.21.3 #3 SMP Thu Jun 28 15:52:05 EDT 2007 i68  
6 GNU/Linux  
bash-2.05b$ /usr/bin/uns query -entityid -l jonojono  
Key: Name: Jonathan Oberheide  
Uid: 157138 Login: jonojono Prev Login: jonojono  
Admin: webunig Prev Admin: webunig Transfer Ack: 1  
Entity-ID: 68914308  
Group: webunig Flag: KLNFX-- From: 20020428 To: 20070428 Login_Ack: 1  
Group: filesys Flag: -----U From: 20020428 To: 20070428 Login_Ack: 1  
Group: caen Flag: -----U From: 20030905 To: 20080903 Login_Ack: 1  
bash-2.05b$
```



Derive Card Number

UMID # 9999 9901

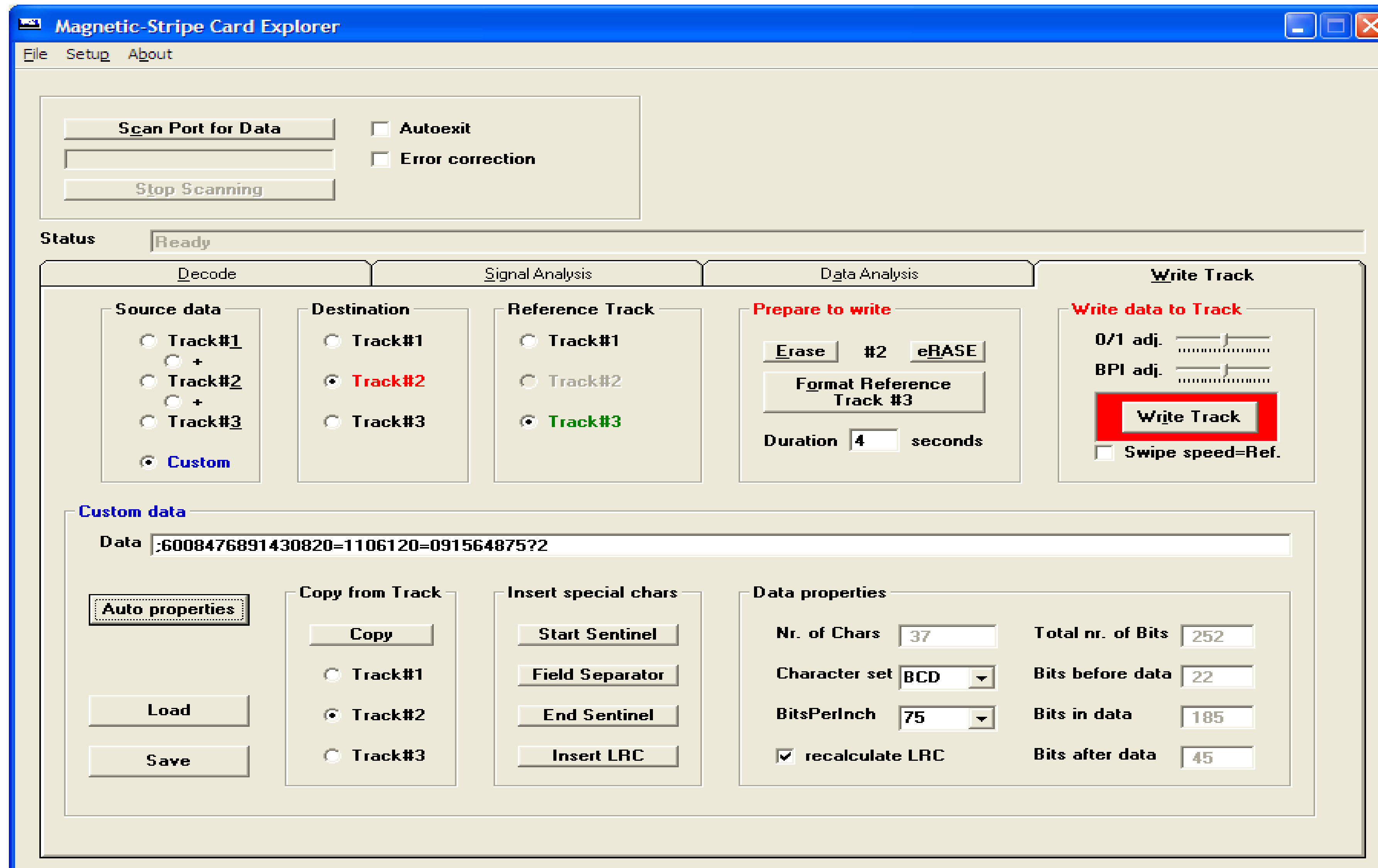
Card # 6008479999990112

CARD NUMBER:

600847 + UMID + Revision + Luhn



Write Magnetic Card



PROFIT!!!



BUT HOW???



Badness

Steal Entree Plus/meal plans

Buy iPods (Ugo's, showcase)

Break into dorms and wreak havoc

- ▶ Frame your enemies!
- ▶ Graffiti incident

Gain physical access as your favorite Umich official

- ▶ marysuec, grue, mabdelah

Steal expensive equipment

- ▶ Target building/facilities managers



More Badness

Arbor Lakes

Chemicals, hospital drugs, etc

- ▶ Hopefully protected by another layer of secure access?

TCF-linked ATM access

- ▶ Mcard acts as ATM card at TCF bank
- ▶ Already compromised “what you have”
- ▶ Fairly easy to obtain “what you know”
- ▶ ATM shoulder surfing + social engineering



Solutions

Vulnerability stems from predictability

- ▶ Don't just read the card number
- ▶ Add extra random data for verification

9 random digits added on track 2

- ▶ $10^9 = 1$ billion tries to brute force
- ▶ Making a card takes ~5 minutes, attack infeasible!

However...

- ▶ Impractical to reissue over 110k cards
- ▶ Gradual replacements for high risk, but no flag day



Agenda

Cosign authentication bypass

Mcard forgery attack

Physical security of CSE

Other things...



CSE Building

- Let's look at a real world example
 - CSE building!
- But, with a hypothetical scenario
 - Plutonium in my office desk drawer
 - Evil terrorist CSE scholars want to obtain it
- How robust is the physical security that protects our valuable research in CSE?
 - Hard balance between an open university building and a locked down supermax facility
- PHOTO TOUR!



Entrance



Vomit Covered Reader?



Inside!



Security Cameras



Security Cameras: DoS

- We don't want to be recorded on video when breaking into CSE!
- Most of these cameras are very simple
 - Take video, fling across network
 - Slim processing capabilities
- But usually have some sort of controlling interface (web, telnet, whatever)
- Simple DoS will make it fall over, drop frames/packets, etc



Security Cameras: 802.11 Attack

- Like ethernet, 802.11 has no link-layer authentication
 - In most situations...
- How to kick someone off WiFi network?
- Just spoof 802.11 Deassociation frame
- Camera will have to reassociate and be unable to transmit video

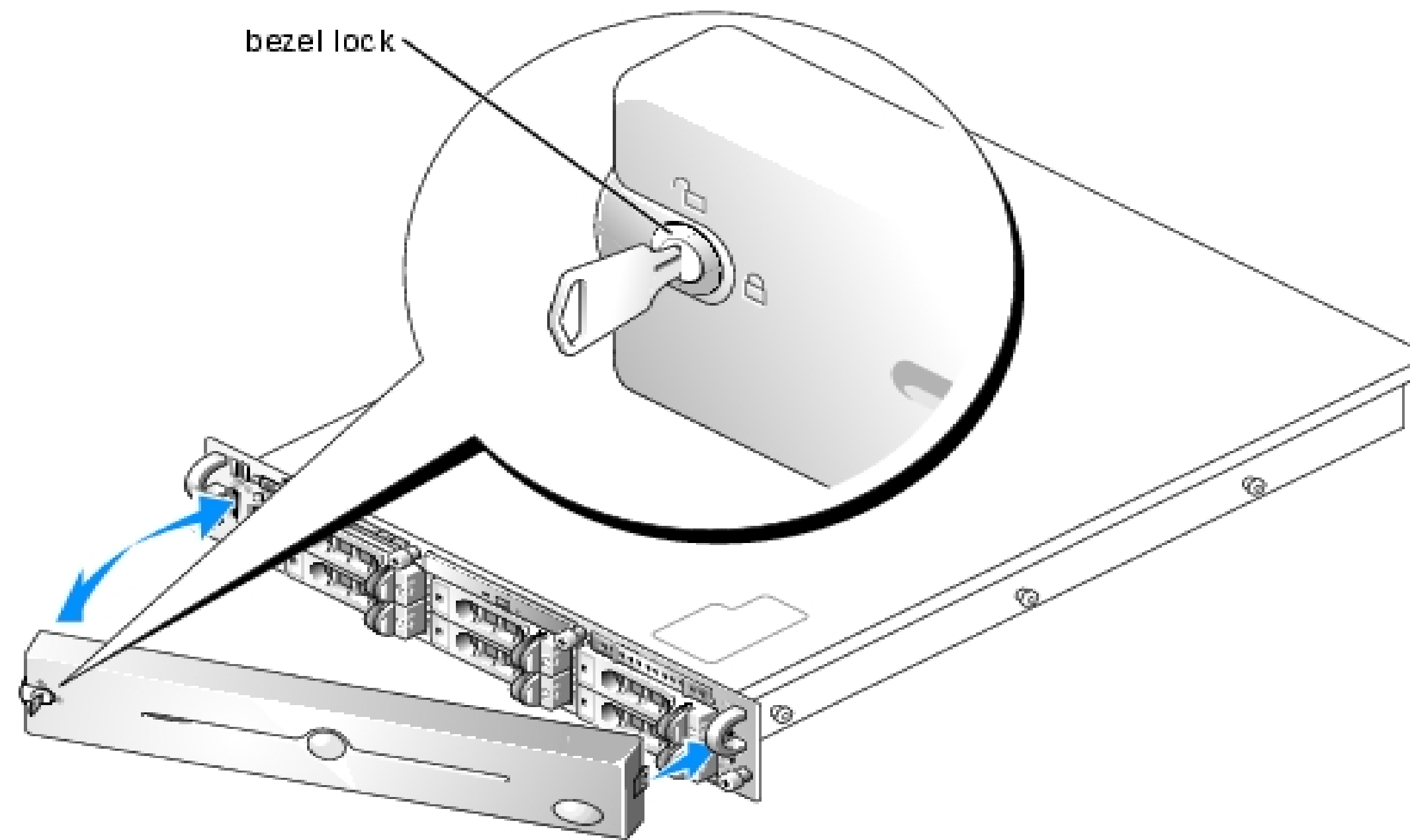


Elevator Access



Elevator Access

- Has special key for maintenance, individual service, lockdown, etc
- Could we pick this lock?
 - Certainly!
- But why pick it when we already have keys to operate it?!?
 - Dell server bezel keys
 - Pretty much any small key can rake it



Elevator Action!



Or, get past locked barriers

- Stairway barrier protected with simple padlock
- Shimming is easy, and fun!
- Bonus points if you use a pop can!



4th Floor



Hallway



Office Lock



Office Locks

- The locks on all our offices (and University-wide locks) are fairly good!
- Schlage Everest with restricted keys



Schlage Everest

- 6 pin - tumbler locks (B145)
- Restricted keys
 - You CAN'T get key blanks
 - Distributed directly by Schlage
 - Also, restricted by patents to prevent 3rd-party blanks
- So master-key creation attacks are foiled
 - Unless we mill our own blanks, but VERY hard
- But can't we still pick the tumbler?



Picking Everest

- Yes, but harder than normal
- The dreaded finger pin!



Picking Everest

- Just need special tool
 - Make your own or buy one for ~\$30



Everest Pick



Open Sesame



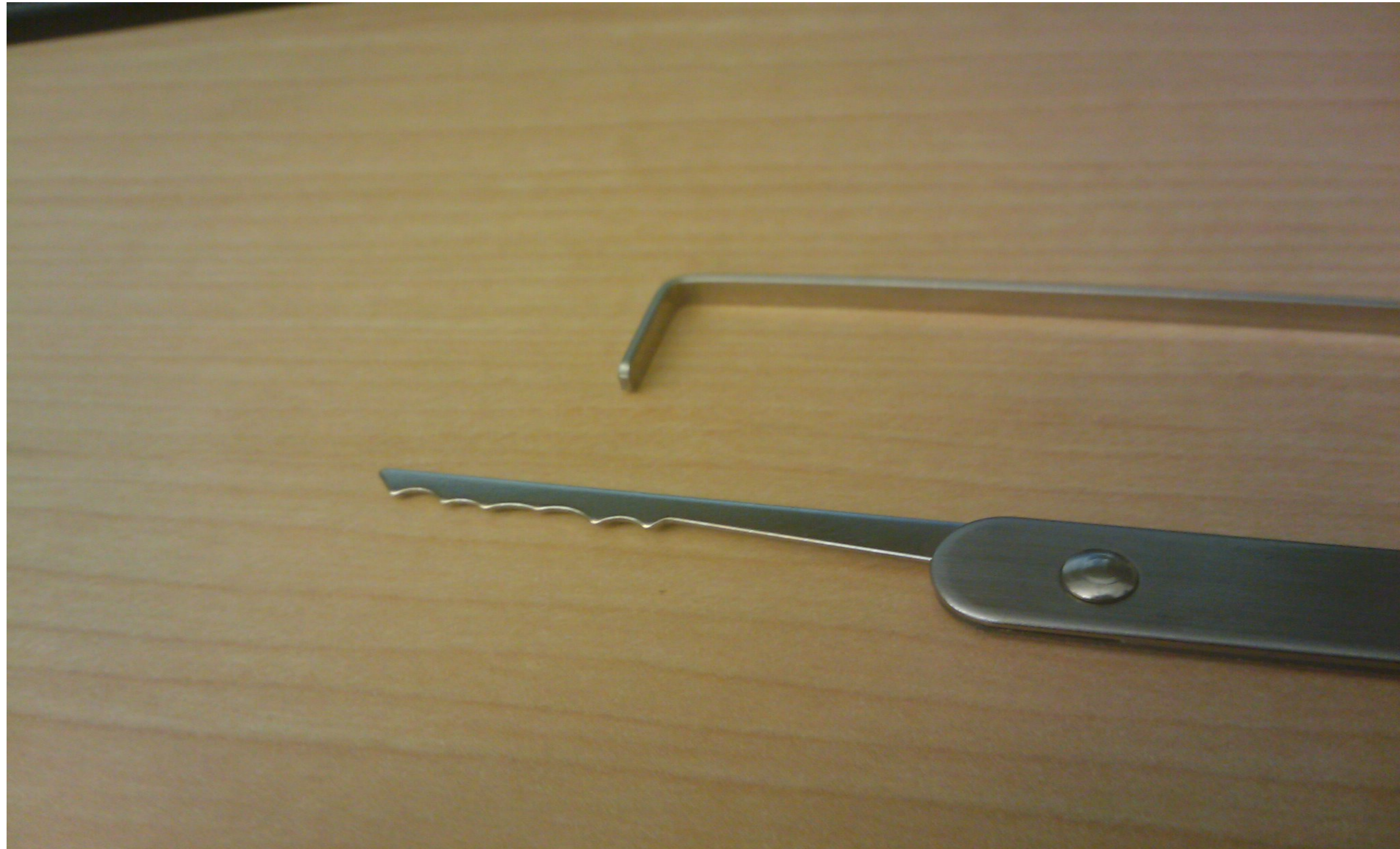
Terrorist Bunker



Locked WMD Office Drawer



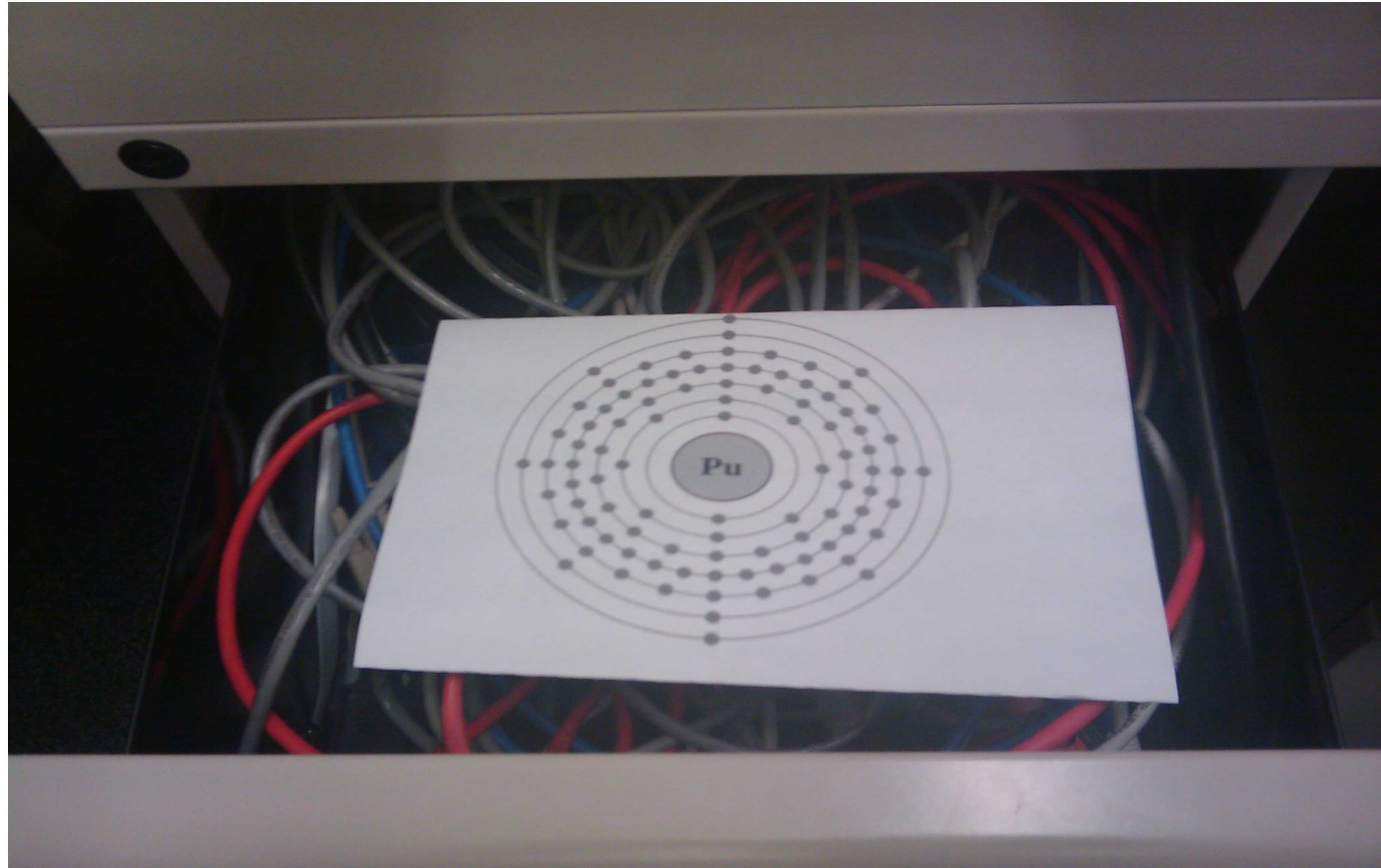
Rake Pick



Boink



Plutonium!



Agenda

Cosign authentication bypass

Mcard forgery attack

Physical security of CSE

Other things...



Duo Tech Talks



^ Car hacking
from these clowns

<http://www.meetup.com/Duo-Tech-Talks/>





ARBSEC 12

Ann Arbor Security Meetup

What
An informal meetup of information security professionals in Ann Arbor. Unlike other meetups, you will not be expected to pay dues, "join up", or present a zero-day exploit to attend.

Where
ARBSEC 12 will be held at the [Tech Brewery](#), located at 1327 Jones Dr, Suite 106.

When
ARBSEC is the first Wednesday of every month. ARBSEC 12 is March 3rd at 6:00 PM. We'll stay until people get tired of hanging out. We're guessing 2-3 hours.

Why
We know about ISSA, SEMISLUG, and SUMIT. Not casual enough. We don't want to hang out in conference rooms. Just a chance to meet other security folks without sitting through a sales pitch.

Contact Us
There is a mailing list: [<announce@arbsec.org>](mailto:announce@arbsec.org). To subscribe, email [Jon Oberheide <jon@oberheide.org>](mailto:jon@oberheide.org) or follow us on [Twitter](#) for announcements.

A2 New Tech Meetup

The screenshot shows the Meetup website interface for the "Ann Arbor New Tech Meetup". At the top, there are navigation links for "Find a Meetup Group" and "Start a Meetup Group", along with "Login", "Sign up", "New Features", and "Help". The main header features the group name "Ann Arbor New Tech Meetup" and a navigation menu with options like "Home", "About", "Meetups", "Ideas", "Members", "Photos", "Discussions", "Money", "More", and "Join us!".

The main content area is divided into several sections:

- Event Details:** "Ann Arbor New Tech March Meetup" is scheduled for March 16th at 6:30 PM. The location is Blau Auditorium, UM Ross School of Business, 701 Tappan Street, Ann Arbor, MI 48104. The event is hosted by "a2geeks".
- Attendance:** 98 Yes, 24 Maybe.
- Who's coming?:** A list of attendees including Charlie Yan (Togo Health), John Paul Narowski (KarmaCRM), Gerry Roston (Wiseman Engine), Brett Wejrowski (HelloRent), and Phil Brabbs (ScoutForce).
- Event Description:** Five presenters will take the stage for ten minutes each, followed by a Q&A session and networking.
- Registration:** A "Want to attend?" section with a "Join us!" button and a "Login" link for existing members.
- Organizer:** "a2geeks" is the organizer, with a list of assistant organizers: Amy Klinke, Dave Brophy, David Bloom, Dug Song, Luis, Roger Rayle, Scott Olson, and Wesley Huffstutter.
- Stats:** The event has 1,021 Technologists, 51 comments, and 20 meetups so far. It was founded on January 28, 2009.
- Our Sponsors:** A section for sponsors is visible at the bottom left.

At the bottom of the page, there is a URL: http://photos4.meetupstatic.com/photos/event/b/2/f/a/highres_7845818.jpeg and a star rating system.

<http://a2newtech.org>



Thank you

QUESTIONS?

Jon Oberheide

jono@duosecurity.com

<https://www.duosecurity.com>

