Detect and Contain

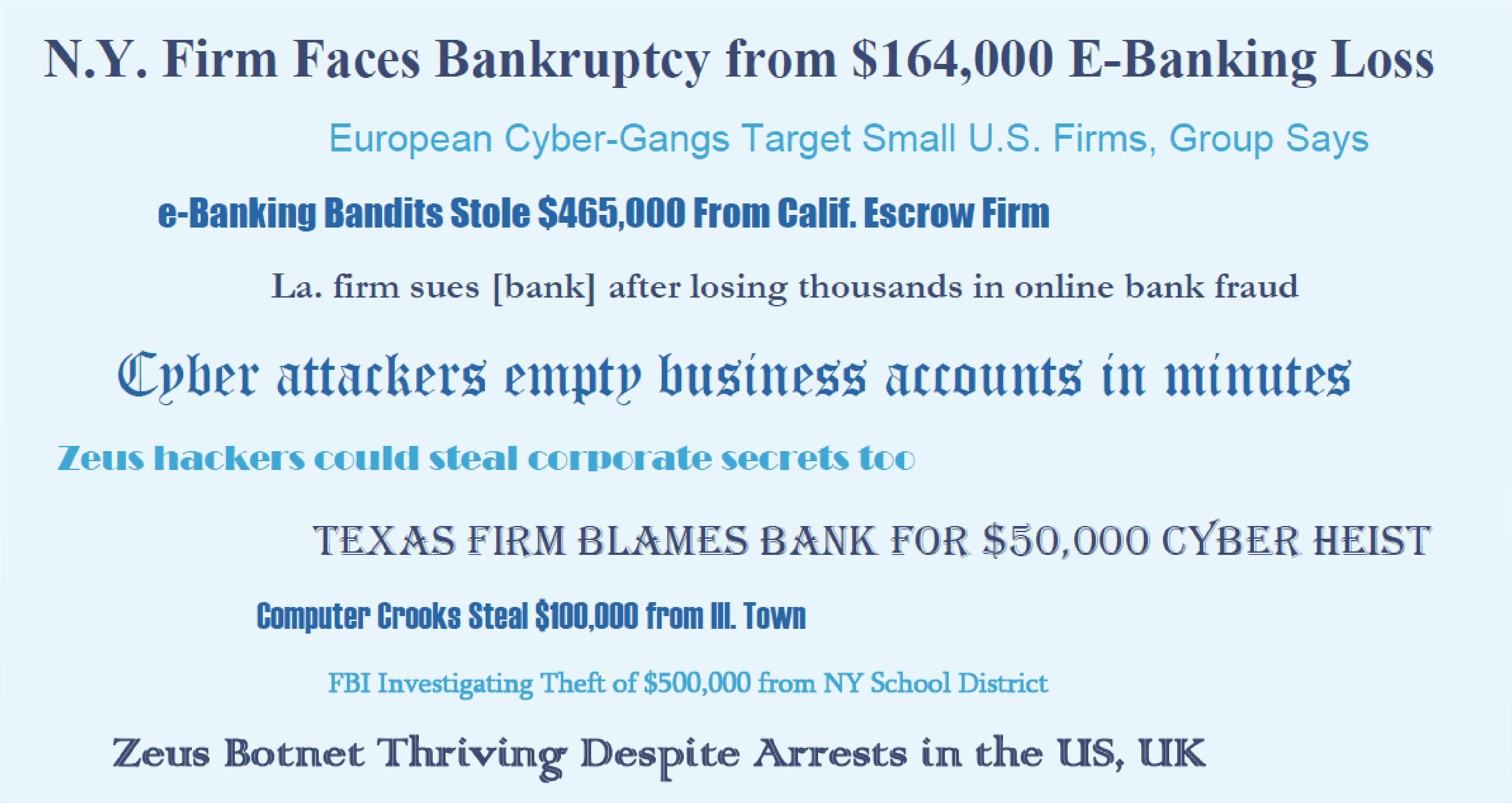
Combating Account Takeover

Robert E. Lee
Twitter: @robert_e_lee

What is Account Takeover?

- Account Takeover is when someone other than the authorized user successfully gains access to the user’s account

Account Takeover in the news



N.Y. Firm Faces Bankruptcy from $164,000 E-Banking Loss

European Cyber-Gangs Target Small U.S. Firms, Group Says

e-Banking Bandits Stole $465,000 From Calif. Escrow Firm

La. firm sues [bank] after losing thousands in online bank fraud

Cyber attackers empty business accounts in minutes

Zeus hackers could steal corporate secrets too

TEXAS FIRM BLAMES BANK FOR $50,000 CYBER HEIST

Computer Crooks Steal $100,000 from Ill. Town

FBI Investigating Theft of $500,000 from NY School District

Zeus Botnet Thriving Despite Arrests in the US, UK

Recent news headlines from *The New York Times, The Washington Post, Computer World, and Krebs on Security. Fraud Advisory for* ***Businesses: Corporate Account Take Over***

How can I detect Account Takeover?

- Organizations require better insights into the behavior of their users. These insights can be gleaned by analyzing ***Who* is doing *What* from *Where***.

- Fully leveraging available event data allows an organization to begin to:
  - Ease the onboarding of new customers
  - Recognize returning customers, offering a more pleasing user experience
  - **Detect and Contain unwanted users & their unwanted behavior**

Part 1 - Actors and Actions

Who is doing *What* from *Where*?

Who: Getting to know your users

- Every application that has a user base should have some notion of a Customer Identification Program (CIP).

- Common data collected during enrollment or within the use of an application include:
  - Name: Title, First, Last, Aliases
  - Contact Information: Address, Phone, Email
  - Payment information: Credit Card, Bank Account
  - State Issued ID: SSN, Driver’s License, Passport

Who: Getting to know your users

- Verification can include simple checks, such as verifying contact details (email, text message, automated voice call, mail, scanned ID card)
  - This step requires an enrollment fraudster to use contact details that they have access to
  - Verification can significantly add to a fraudster's time and monetary costs
  - Verification can provide better nodes for Link Analysis

- External services can Validate identity by comparing the collected identity data to the public record:
  - Name <-> Address, Name <-> Phone Number, Address <-> Phone Number, Name <-> State Issued ID, etc
  - This step can reduce the number of enrollments with fictitious identities
  - You may also choose to block enrollment from certain sets of identity data, such as identities that are known to have been previously associated with account takeover, or identity theft

What: What are my users doing?

- All of the interactions with your application are discrete events worthy of logging and analysis

- Events of particular note include:
  - Enrollment
  - Authentication
  - Profile/contact changes
  - High Risk/Sensitive transactions

What: What are my users doing?

- To enable Link Analysis and Behavior Monitoring capabilities, it is imperative to have complete log records:
  - ***Who***: User ID, or Personally Identifiable Information (PII)
  - did ***What***: Type of transaction
  - ***When***: Time stamp for event
  - from ***Where***: IP address, Location, Device ID
  - with what ***Result***: Success, Failure
  - and in what ***Context*** (session/role)

W*here:* From where are my users accessing?

- Two common methods of determining *Where* involve the IP Address the user appears to be coming from, and the Device ID assigned to a user’s machine.

- Location:
  - There are many services available that attempt to associate an IP Address with an approximate physical location
  - Through HTML-5 or other methods, depending on device type, it may be possible to query the device for current GPS location

W*here:* From where are my users accessing?

- There are several drawbacks to using an IP address derived *Where*. Chiefly:
  - IP Addresses are often assigned dynamically for a short period of time
  - A single IP Address can represent many devices
  - A single device may cycle between multiple IP addresses; location change, mobile device, a device proxy hopping
    - It is trivial to change which IP address a session appears to be coming from through use of proxies

For these reasons, it is desirable to supplement IP Address based identification with a more precise method. Modern Device Identification technology enables organizations to uniquely identify a device, and associate the event data with an individual device.

Where: From where are my users accessing?

- A Device ID should:
  - Accurately identify a unique device in a way that is resistant to manipulation:
    - http://samy.pl/evercookie/ -- One part Tag
    - https://panopticlick.eff.org/ -- One part Fingerprint
  - Allow for the recognition of a returning device
  - Not require active participation of the user
  - Have checks for signs of proxy use
  - Be included in the event logs
    - Once certain user behavior is observed as fraudulent, can link to other sessions from the same DeviceID

Part 2 – Common failures & Improvement Ideas

"*Failure is success,* ***if*** *we learn from it*"

--Malcolm Forbes

Common Methods of Account Takeover

- Enrollment Fraud
  - – Attacker gets to choose their own password, and specify phone number

- Knowledge Challenge Compromise
  - – Automated brute force tools: JTR, Hydra, etc
  - – Capture: Phishing, Key logging, Advanced Malware (MitM/MitB)
  - – Weak “Forgotten Password” flow
  - – Password reuse from previously compromised site

Common Methods of Account Takeover

- Possession Challenge Compromise
  - Leaked token seed values
  - Interception of data destined for a physical device
  - Interception of the physical device

- Session Hijacking
  - Cookie reuse
  - MitB

Identity

- Collect enough data about your user to uniquely identify them
  - User ID alone is insufficient for sensitive applications
- Verify the user has access to the contact details they’ve enrolled with:
  - Adds time and monetary costs for fraudster
  - Gives weight to nodes used in link analysis
- Use a public record Validation service
  - Helps ensure identity data is real and self consistent
- Device ID **and** IP Address are part of identity
- Allow users to name their devices
  - Reference device name and location data in user communication

Authentication - Passwords

- Passwords are shared secrets that should only be known by the user and the authentication system
  - Passwords should be difficult to guess, yet easy to remember
  - If you use the same password on multiple sites, it’s no longer a secret

- Password policies often include:
  - An alphabet inclusion requirement (roman alphabet, numbers, special characters, case enforcement)
  - A minimum length requirement
  - An auto-lock out (hard lock, or incrementally time scaled lockout)
  - An expiration
  - A password rotation policy (can not use the same password as previous X).

Authentication - Passwords

- The relative strength of a policy can be calculated:
  - Alphabet requirement: roman alphabet (not case sensitive), numbers – 36 characters
  - Length: 6
  - Auto-lock out for 24 hours after 10 consecutive failed attempts (from any IP address) – rate of 10 tries/day
  - Expiration: 365 days

  - P=(10*365)/36^6, or 0.000168% of the entire password space can be exercised.
    - http://prezi.com/u1kpvimvoiwd/password-strength/

Authentication - Passwords

- Cap brute-force attempts
  - Lock after X consecutive failed attempts, and/or Y total failed attempts, in Z timeframe
- Don’t require the user to change their password too frequently
  - Good passwords are hard to remember
- Encourage passphrases over passwords
  - Length is crucial to increasing brute force time

Authentication - Passwords

- Enforce a sane entropy requirement
  - ~2.5 bits per byte seems reasonable
  - http://www.fourmilab.ch/random/

- Enforce a dynamic wordlist check
  - Make sure not more than N users or N% of users in the system are using the same password
  - http://research.microsoft.com/pubs/132859/popularityiseverything.pdf

Authentication – Challenge Questions

- Challenge questions often elicit answers that are quasi public knowledge
  - If the answer is known by more than the user and the authenticating system, then it is not valid for use in authentication

- Challenge questions are not subjected to the same complexity requirements as passwords
  - Brute forcing answers to challenge questions can often be easier than brute forcing passwords

Authentication – Challenge Questions

- Passwords and answers to Challenge Questions are both knowledge based challenges
  - An authentication system with two challenges of the same factor is still Single Factor Authentication
  - For a MFA system, consider replacing Challenge Questions with a *possession* based challenge

- Challenge Questions offer poor usability
  - Users often forget the answers

Authentication – Challenge Questions

- Static Challenge Questions are being phased out in favor of out-of-wallet/dynamic questions
  - Pro: Dynamic questions require no enrollment
  - Pro: If an attacker passively collects answers (phishing, mitm, etc), less useful for next session
  - Con: Can often be answered by anyone close to person; family members, ex-gf/spouse, close friends, someone with access to credit history
  - Con: If sourced by public record databases, by definition, not a secret, therefor invalid for authentication
  - Con: Sometimes the public record data is wrong

Authentication – Email

- Email is not something the user *knows*, *has*, or *is*
  - Email can therefor not be considered a factor in authentication
  - Still, Email can be used as part of the identity profile

- Unless otherwise provided for (SMIME, PGP, etc), email is not encrypted
  - Email is unsuitable for sending anything that requires confidentiality
  - Passwords, even One-Time Passwords require confidentiality
  - Special links that provide for privileged access require confidentiality too

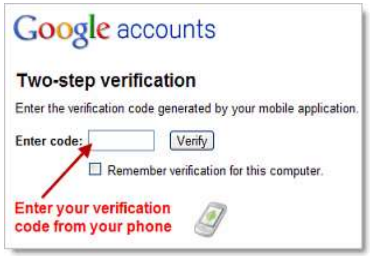
Authentication - MFA

	Something you **KNOW**	Something you **HAVE**	Something you **ARE**
ATM	**Pin**	**Card**	
Application	**Password, Challenge Question**	**Token, Phone**	**Fingerprint, Facial Recognition**

Users expect protection

- Examples:
  - Google
  - Facebook
  - World of Warcraft
  - PayPal

- If free email, social media, and video games have MFA, **why doesn't your sensitive application?**



Google accounts

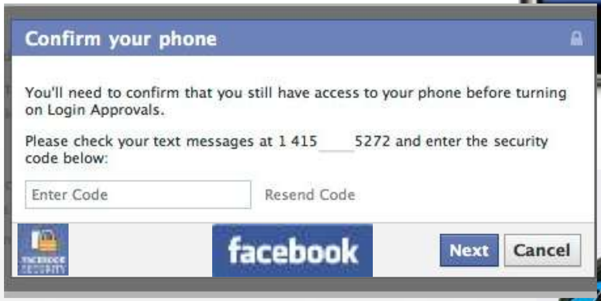
Two-step verification

Enter the verification code generated by your mobile application.

Enter code: [ ] Verify

□ Remember verification for this computer.

Enter your verification code from your phone



Confirm your phone

You'll need to confirm that you still have access to your phone before turning on Login Approvals.

Please check your text messages at 1 415 5272 and enter the security code below:

Enter Code | Resend Code

Authentication – OTP

- A One Time Password is a knowledge challenge that is often associated with a physical object, and often time bound

- Token based OTP requires secrecy of seed value
  - https://en.wikipedia.org/wiki/SecurID

Authentication – OTP

- OTP’s communicated to a user out-of-band (SMS, Voice) are often still collected in-band
  - Unless the user *responds* out-of-band, this is still an in-band authentication

- Typical OTP implementations rely on the application to provide authentication context to user
  - In a MitM/MitB/Phishing scenario, user can be socially engineered into giving an attacker their OTP credential

Authentication – Cookies

- Cookies are digital artifacts that can be copied and reused
  - If two or more people can simultaneously “have” the item, it is invalid for *possession* based authentication

- FFIEC definition of Complex Device Identification
  - Have the value in the authentication cookie change each session
  - Restrict use of cookie to systems with identical fingerprint

Authentication – Forgotten Password

- Forgotten Password features often utilize weaker authentication controls than normal authentication flow
  - In many MFA implementations, Forgotten Password relies solely on a possession challenge to reset the knowledge credential, rendering the MFA solution in actuality, Single Factor Authentication
  - Especially worrisome if your “possession” challenge is mistakenly using Email

- Resetting a password is a high risk transaction
  - Protect it 1st with a valid *possession* challenge
  - Supplement with alternate *knowledge* challenge

Assume Compromised System

Web Browser

Trusted Path (OOB)

User Interaction – Alerting

- Users should be notified when important changes are made

- If alerted of a sensitive change they did not make, users will naturally contact you

Context Aware, Out-of-Band

Blind Authentication	Context Aware Authentication
1. User starts to log into online banking with Username and Password 2. User is prompted for an OTP 3. User Enters OTP into web browser to complete authentication	1. User starts to log into online banking with Username and Password 2. User receives a message OOB showing an authentication attempt, complete with the username, device name, IP address, location, and date/time of attempt. 3. User responds OOB "approve" to complete authentication
1. User initiates a wire for $10,000 to normal vendor 2. User prompted for an OTP 3. User enters OTP 4. Days go by 5. User discovers their wire amount was changed to $1,000,000 and sent to an unknown account 6. Possible lawsuit filed; Bad PR	1. User initiates a wire for $10,000 to normal vendor 2. User receives a message OOB showing a wire attempt for $1,000,000 to an unknown account 3. User denies the transaction and reports the transaction as fraud 4. Company helps user clean infected device

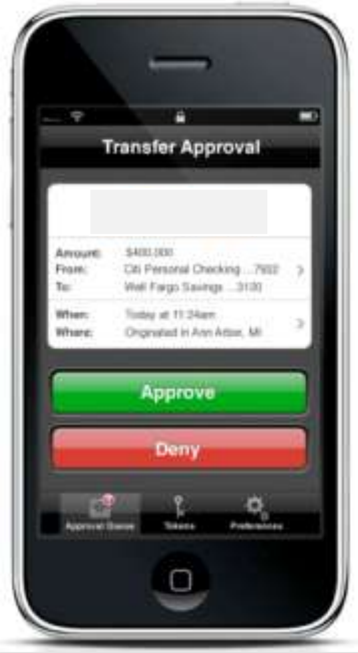
Armed with contextual awareness, users can make intelligent authentication decisions
Responding OOB reduces chance of giving authentication credential to attacker

Authentication – Text/Voice

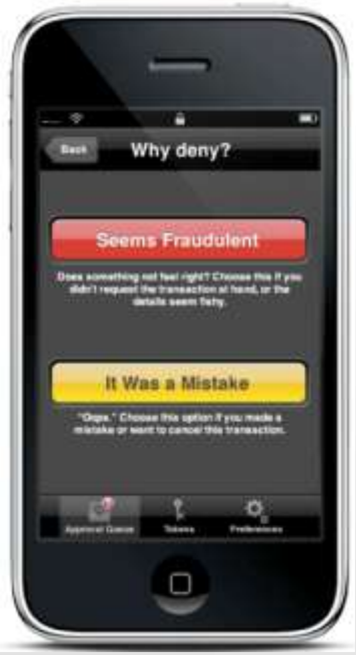
- Text/Voice based authentication can be intercepted
  - Some platforms (blackberry, android, etc) allow for running unsigned code and have been targeted for special SMS interception malware: https://www.google.com/#q=zitmo
  - Calls can be forwarded http://www.snopes.com/inboxer/scams/forward.asp

Authentication – Trusted Path

- Ideal to use native applications with signed, encrypted, two-way messaging
  - Better Security
  - Better Usability



Amount: $400,000
From: Citi Personal Checking ...7932
To: Well Fargo Savings ...3130
When: Today at 11:24am
Where: Originated in Ann Arbor, MI



Does something not feel right? Choose this if you didn't request the transaction at hand, or the details seem fishy.

"Oops." Choose this option if you made a mistake or want to cancel this transaction.

Example Vendor:
http://duosecurity.com

Link Analysis

- Many organizations are not performing thorough root cause analysis or link analysis for confirmed security incidents
  - Unaware of what control(s) failed; stay blind of what to fix
  - Unaware of true scope of incident; unable to reach out to other affected users, business partners

Part 3 – Putting it all together

"*The trick to forgetting the big picture is to look at everything close-up*"

--Chuck Palahniuk

Detect and Contain: Building Blocks

- Ensure log completeness
  - *Who* did *What*, *When*, from *Where*, *Result*, *Context*

- Identify high risk transactions in your application
  - Enrollment
  - Authentication
  - Profile/contact changes
  - High Risk/Sensitive transactions

- Determine appropriate authentication assurance requirements to perform transactions
  - Interesting implications for SSO
  - Deploy appropriate controls for risk tolerance

Detect and Contain: Building Blocks

- Set up authentication checkpoints within your application to protect high risk transactions
  - May be too onerous to challenge users every time at checkpoints
  - Can scale back challenges initially based on thresholds (challenge on money movement over X amount)

- Ensure healthy incident response program
  - Don’t just track incident count; where possible, identify which control failed, and how it failed
  - If incident is associated with a vulnerability, track incident count against known vulnerability
  - Perform Link Analysis to see which other sessions/actors are related to incident

Detect and Contain: Finishing Touches

- Deploy a real-time risk scoring system:
  - Automatic learning from the event data stream
  - Should also learn from manually tracked incidents
  - Multiple models: User vs Self and User vs App Population
  - Should take all pieces of Identity, and reputation of identity into consideration
- Integrate application checkpoints with risk scoring system (RBA)
  - Challenge based on risk score, and risk tolerance
  - Delivers the best blend between security and usability

Detect and Contain: Finishing Touches

- Work with Data Scientists
  - You can only write pattern matching rules once you know what you’re looking for
  - To stay ahead of the incident curve, invest in people and technology (Big Data) that can help you identify new trends

Summary

- To Detect and Contain Account Takeover:
  - Understand *Who* your users are, *What* they’re doing, *When* and from *Where* they do it
  - Deploy higher assurance authentication controls
  - Strive for real-time action based on your event stream

Q/A

- Special thanks to:
  - Pete Herzog, ISECOM
  - Jeremiah Grossman, WhiteHat Security
  - Mike Eynon, Silver Tail Systems
  - Christian Ruvalcaba, Intuit
  - Tom Pigoski, Intuit
  - Scott Collins, Intuit

Robert E. Lee
Twitter: @robert_e_lee