

Threat Modeling for Game Developers

Stephen Beeman

Proprietor, Gizmocracy

stephen.beeman@gizmocracy.com

Game Developers Conference® Online 2011
October 10-13, 2011 | Austin, TX
www.GDCOnline.com

GDC
11
Online

Not My First Rodeo

- Programming games since 1978 (TRS-80, represent!)
- Wrote my first LAN game in 1984
- Wrote my first online game for Microsoft in 1996
- Director of Online Technology for Origin
- Lead program manager for server security and infrastructure, MSN Games (Windows/C#)
- System architect for Superhero Squad Online (Linux/Python & Java)

The Three Questions

- 1.“What’s the problem?”
- 2.“What will it take to make it go away?”
- 3.“What happens if I just ignore you?”

Definition

- Threat modeling is a process by which a system is methodically analyzed to find, evaluate and mitigate vulnerabilities.
- Threat modeling looks at your system the way an attacker does.

Definition

- A process by which a system is methodically analyzed to find, evaluate and mitigate vulnerabilities.

Definition

- A process by which a system is methodically analyzed to find, evaluate and mitigate vulnerabilities.

Definition

- A process by which a system is methodically analyzed to find, evaluate and mitigate vulnerabilities.

Definition

- A process by which a system is methodically analyzed to find, evaluate and mitigate vulnerabilities.

Definition

- A process by which a system is methodically analyzed to find, evaluate and mitigate vulnerabilities.

Definition

- A process by which a system is methodically analyzed to find, evaluate and mitigate vulnerabilities.

The Process

- a) Enumerate threats
- b) Construct threat trees
- c) List vulnerabilities
- d) Calculate risks
- e) Take action
- f) Repeat

Enumerate threats

- Threats = exploits x assets

Exploit taxonomy: STRIDE

- Pretending to be another user: spoofing
- Modifying data outside of normal usage: tampering
- Erasing the history of an action: repudiation
- Reading data outside of normal usage: information disclosure
- Preventing the system from functioning: denial of service
- Getting permission to perform forbidden actions: elevation of privilege

Example: Auth Cookie

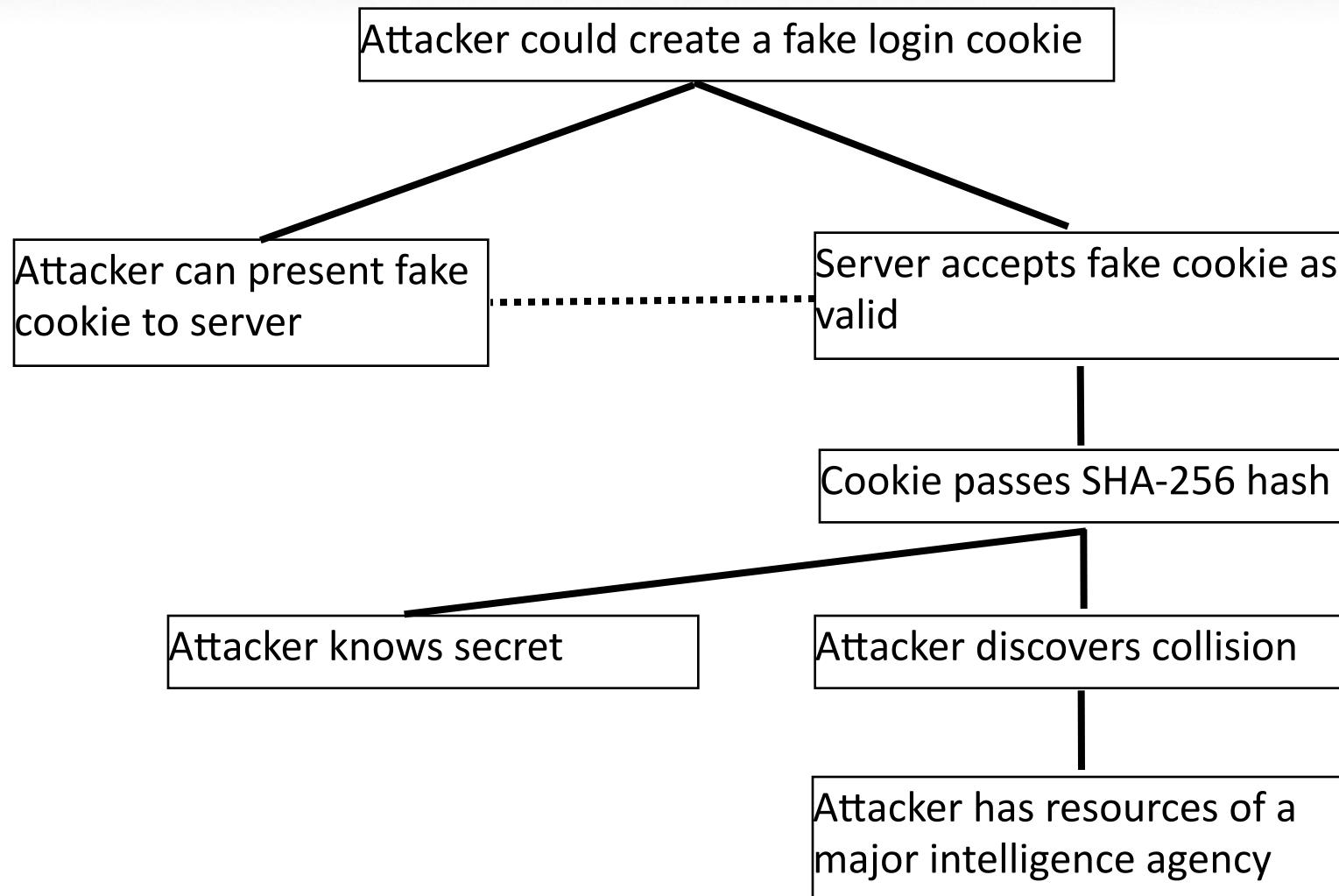
- Spoofing: Create a valid cookie for a real user, or for a fake one
- Tampering: Modify part of a valid cookie
- Repudiation: ...
- Information disclosure: Grab another user's cookie
- Denial of service: ...
- Elevation of privilege: Add permissions tokens to a cookie

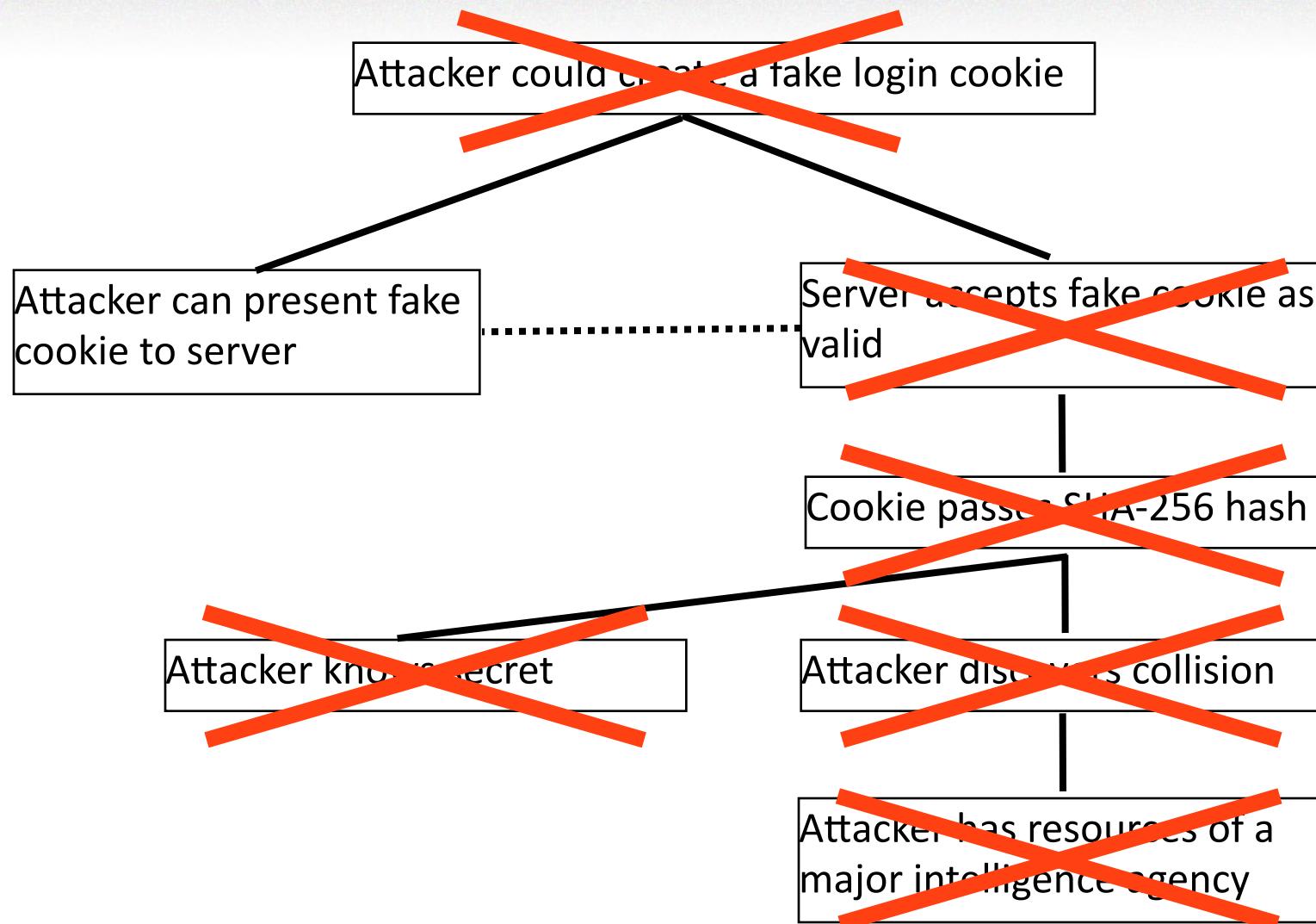
Example: Auth Cookie

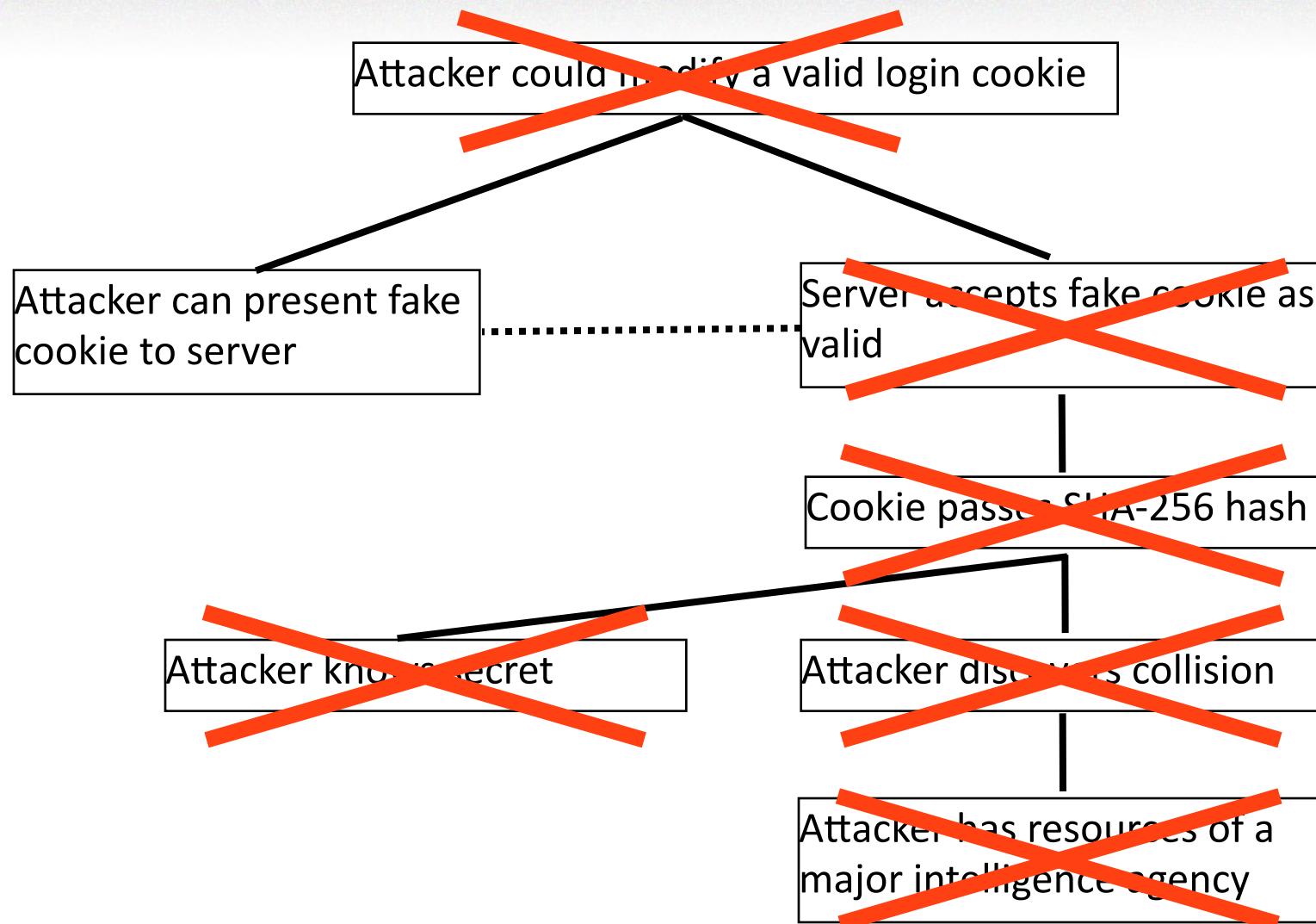
- Spoofing: Create a valid cookie for a real user, or for a fake one
- Tampering: Modify part of a valid cookie [S]
- Repudiation: Modify a valid auth cookie [T]
- Information disclosure: Grab another user's cookie
- Deauthentication: ...
- Elevation of privilege: Add permissions tokens to a cookie

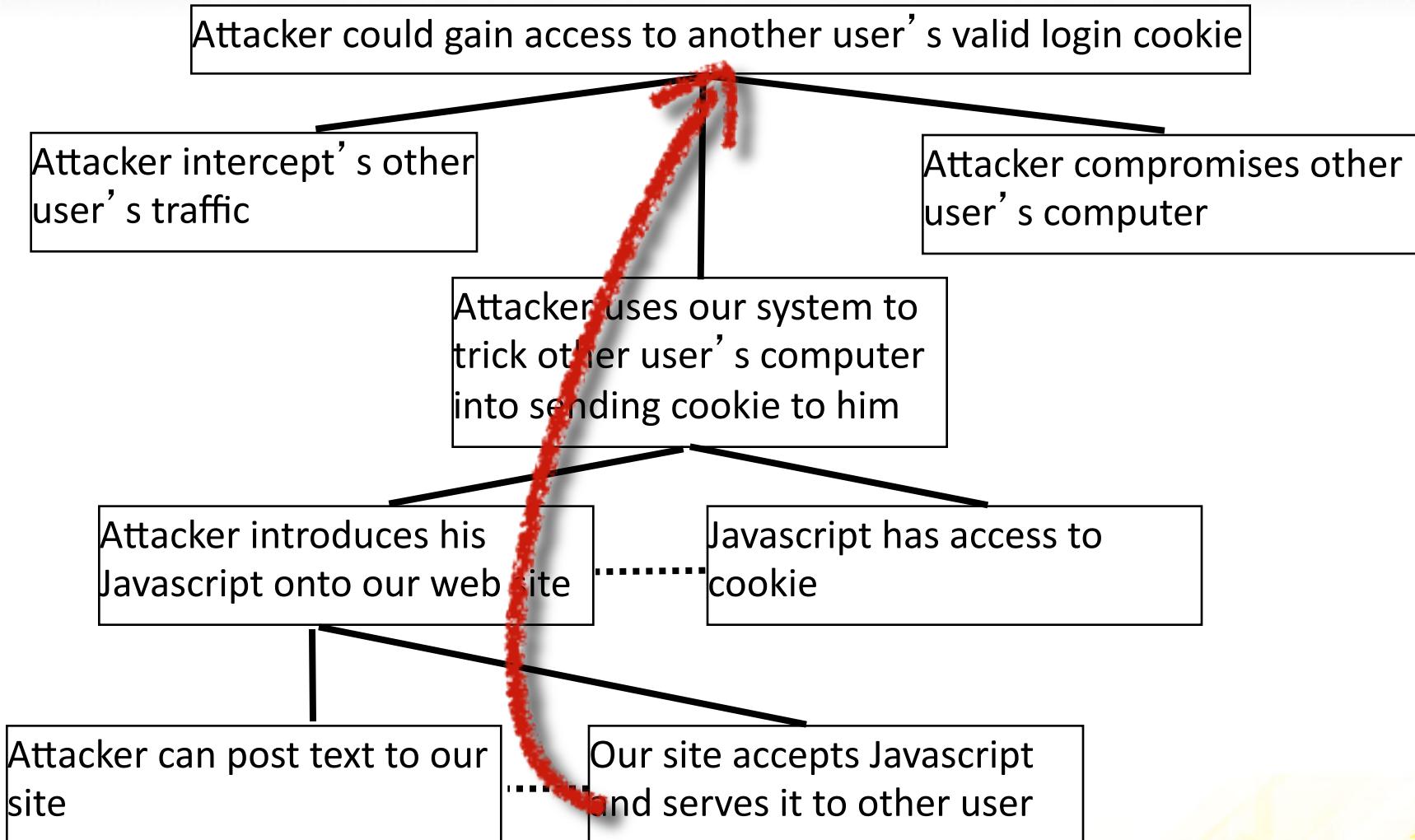
Construct threat trees

- A threat tree is a diagram that describes what must be true for an attacker to carry out the specified threat.
- The topmost node is the threat itself. Each child is a condition that, if true, means the threat can be carried out. Each child in turn has its own children.







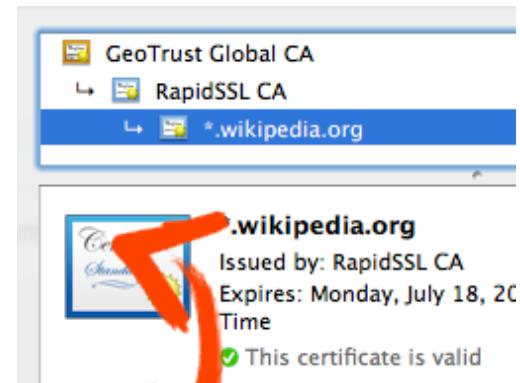
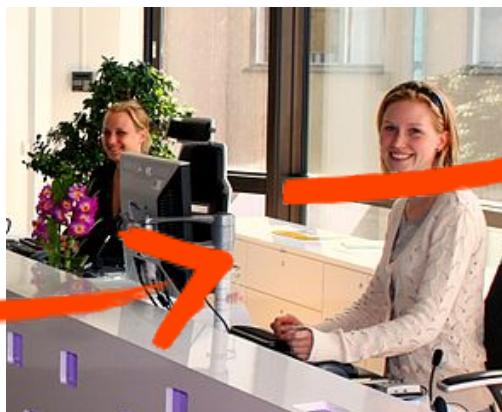
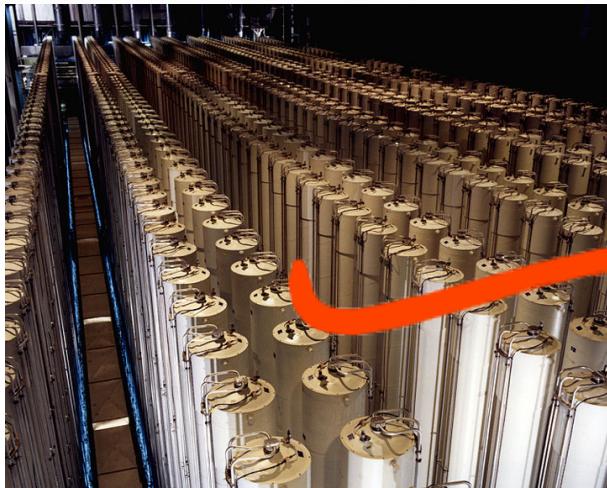


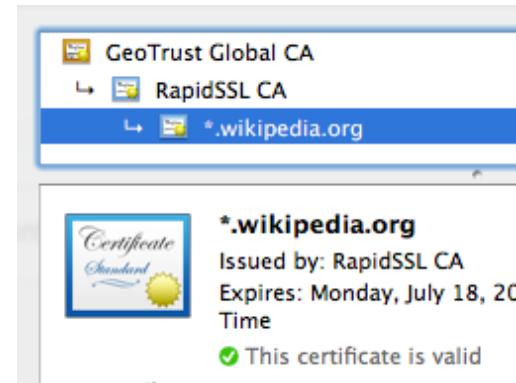
Calculate risks

- A vulnerability is a threat whose threat tree contains one or more valid paths from bottom to top.

That doesn't mean it needs to be mitigated!

- Seriously:
 - There is no such thing as perfect security.





Risk = likelihood x cost

- The fundamental question about any vulnerability is “What happens if we just ignore it?”
- The risk posed by an event is its likelihood multiplied by the cost that event would impose if it took place.
-

Risk rating: REAL

- Reward: What's the attacker get out of it?
- Effort: How little does the attacker have to work?
- Audience: How many people will be affected?
- Level of skill: How many attackers have the skill required to carry out the attack?
- Assign each a value from 1 to 10, multiply them all together and move the decimal two to the left, for a value from 0.01 to 100.0.

Attacker could gain access to another user's valid login cookie

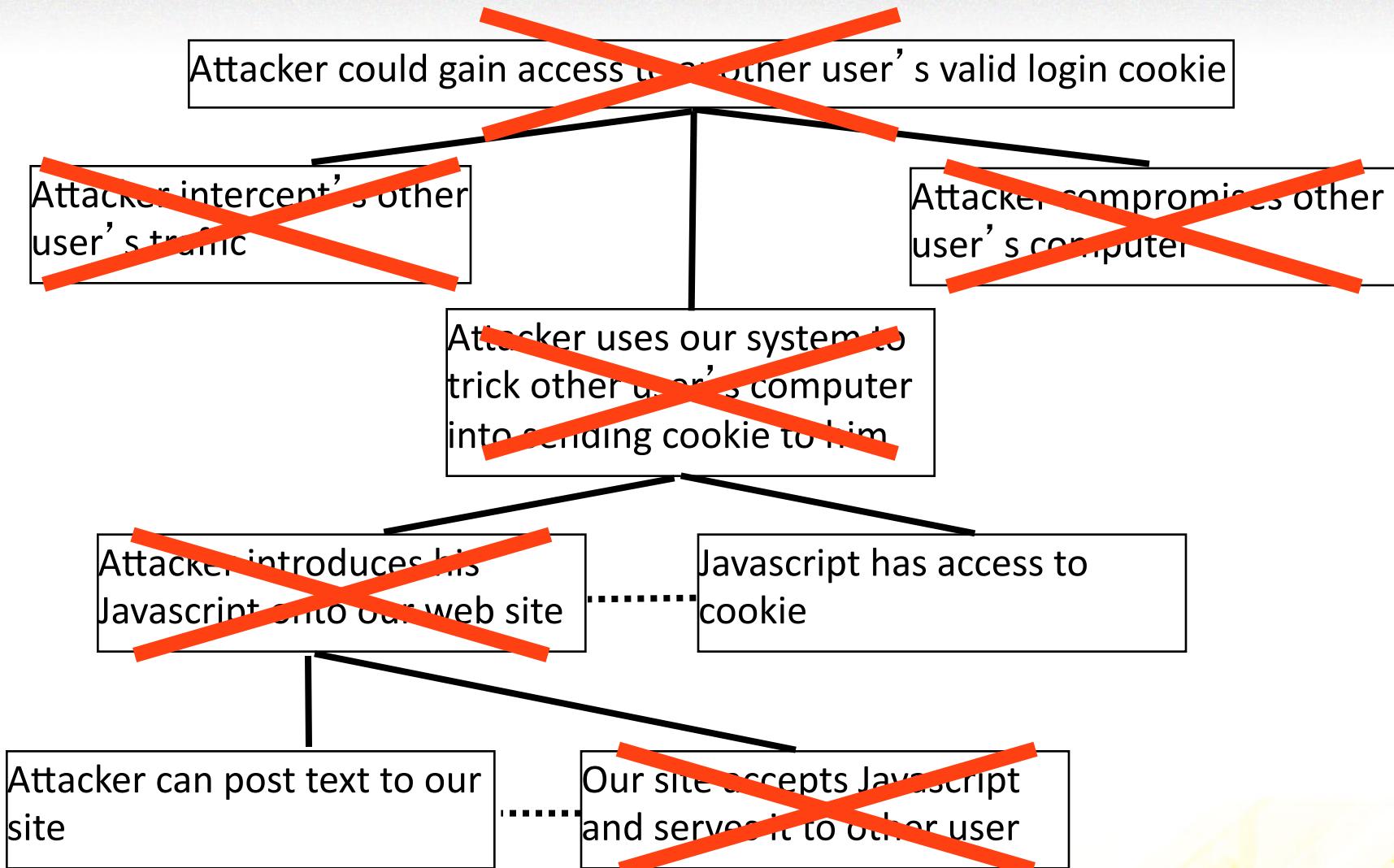
- Reward: 10.
- Effort: 8.
- Audience: 10.
- Level of skill: 8.
- Total risk: 64.0.

Take action

- Set your security bar, the risk rating above which you will act on a vulnerability.
- Enter the vulnerabilities above that bar into your bug database. The ones below that bar go into your backlog.
- Act on the vulnerabilities that cross your bar.

Mitigating vulnerabilities

- Two choices:
 - 1.Change the circumstances so that paths through the threat tree are closed off
 - 2.Change the risk variables so that the vulnerability falls below the bar



Change the risk

- This is perfectly legitimate!
- Reduce the reward!
- Increase the effort!
- Limit the audience!
- Raise the skill level!

Reduce the risk enough, and you can call it a
“real” mitigation

Security through design



Security through education

The screenshot shows a forum post titled "(Sticky Locked) Password Security Information" by Auryk v Support Forum Agent. The post discusses the importance of using different passwords for different accounts to prevent account theft. It includes links to more information on password security and account security.

(Sticky Locked) Password Security Information

Auryk v Support Forum Agent

We have been helping players deal with account theft for years now, and unfortunately, roughly a third of players make a very basic security mistake: using the same password for all of their security needs.

If you are serious about protecting your account and your personal security, your Battle.net password should be different from your email account password – or other personal passwords for that matter!

No one wants account thieves rooting around in their personal email, address book, and contact lists. Too often we see thieves breaking in to this information because their target has used the same password across multiple types of accounts. Not only can this give thieves access to your account, it can lead to compromises far outside of Battle.net as well.

It's immensely important that everyone use separate passwords for separate applications, including games. Secure passwords have both numeric and alphabetical values, and are usually at least 10 characters in length.

For more information on password security, please click here: http://us.blizzard.com/support/article.xml?locale=en_US&articleId=20574

For more information on account security, click here: <http://us.battle.net/en/security/>

Security through business development

BLIZZARD STORE NORTH AMERICA
[Change Region >](#)

Home PC Games Tabletop Games Apparel Books Collectibles More Products



Blizzard Authenticator Cataclysm Edition (United States Only)

Battle.net Authenticator for use with your World of Warcraft account or Battle.net account. This newest design features the Goblin and Worgen art.

Protect your World of Warcraft account with industry leading account security - introducing the Battle.net Authenticator! The Battle.net Authenticator is designed as a supplemental authentication method for your World of Warcraft account, giving you the security of Two-Factor authentication. Each time you log in using the Battle.net Authenticator you are provided with a unique, one-time use password to use in addition to your regular password. Log in with both and you can rest easy knowing that your account is now even more secure from malicious attacks such as keyloggers and trojans.

- Simple and easy to use – press one button to display the digital code. Setup of the token is simple and takes only a few minutes.
- Small and convenient – take your token to wherever you play World of Warcraft and know that your account is secure.
- Tough and durable – lasts for years and replacement is easy.
- Provides for the highest account security available in the game industry today.

★ Limit 2 units per order and 6 units per month. Battle.net Authenticator not for resale.

DOWNLOAD THE
BATTLE.NET MOBILE AUTHENTICATOR

Repeat!

- a) Review and update threats
- b) Review and update threat trees
- c) Review and update mitigations
- d) Identify new vulnerabilities
- e) Recalculate risks and adjust security bar
- f) Address more vulnerabilities

Responsibilities

- Programming: Review threat model for validity and currency; create mitigations
- Production: Manage process and documentation; review risks and adjust security bar
- Design: Create mitigations
- Test: Verify mitigations; maintain security bug list.
- Operations: Review threat model for validity and currency; create mitigations.
 - Ops can and should be doing their own threat modeling of data-center and OS-level security.

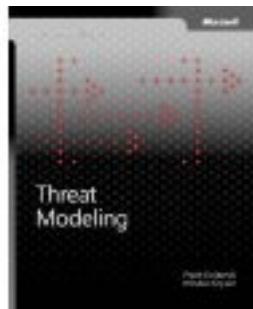
Responsibilities

- Customer service: Review risk ratings; prepare for known vulnerabilities; identify new vulnerabilities in the wild.
- Business development: Create mitigations; review and adjust risk ratings; prepare for known vulnerabilities.
- Community management: Create mitigations; review risk ratings; prepare for known vulnerabilities; identify new vulnerabilities in the wild.

What We've Learned

- a) Enumerate threats
- b) Construct threat trees
- c) List vulnerabilities
- d) Calculate risks
- e) Take action
- f) Repeat

Next Steps



Threat Modeling
Frank Swiderski and Window Snyder
Microsoft Press

<http://www.microsoft.com/security/sdl/>

A screenshot of the Microsoft Security Development Lifecycle (SDL) website. The page has a green header bar with the title "Microsoft® Security Development Lifecycle". Below the header, there's a video player showing a video titled "Write more secure software." A call-to-action button "Watch the video." is visible. The main content area features a diagram of the SDL process with seven phases: Training, Requirements, Design, Implementation, Verification, Release, and Response. A quote from Forrester Research states: "Those practicing SDL specifically reported visibly better ROI results than the overall population." To the right, there are two columns: "Why Adopt the SDL?" which lists benefits for developers and business owners, and "Resources" which links to various documentation and tools.

Other search terms: “attack modeling”, “risk modeling”

stephen.beeman@gmail.com

Q & A