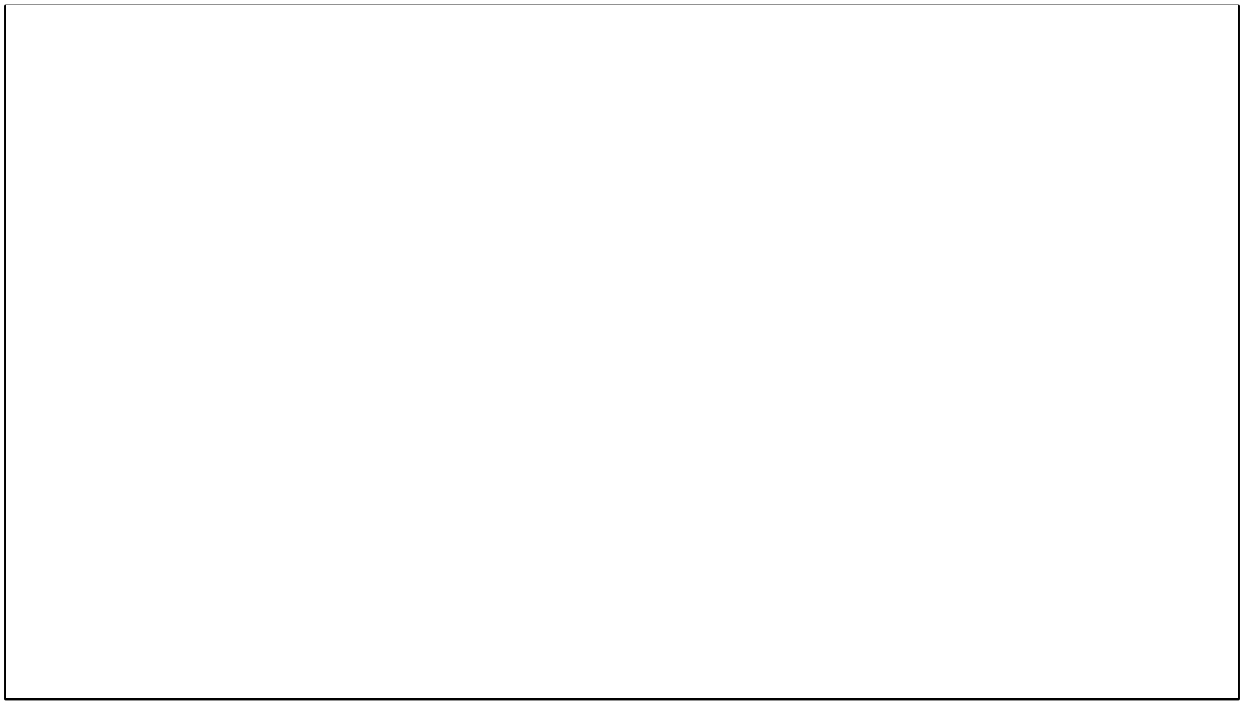
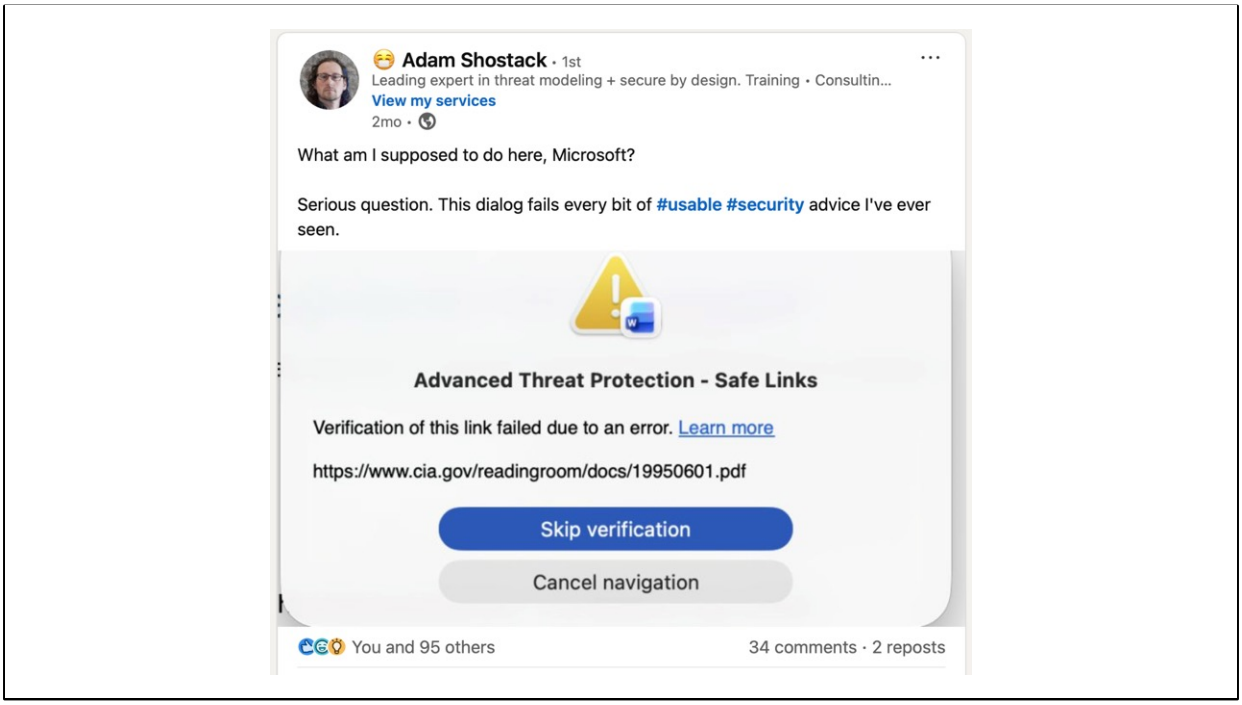
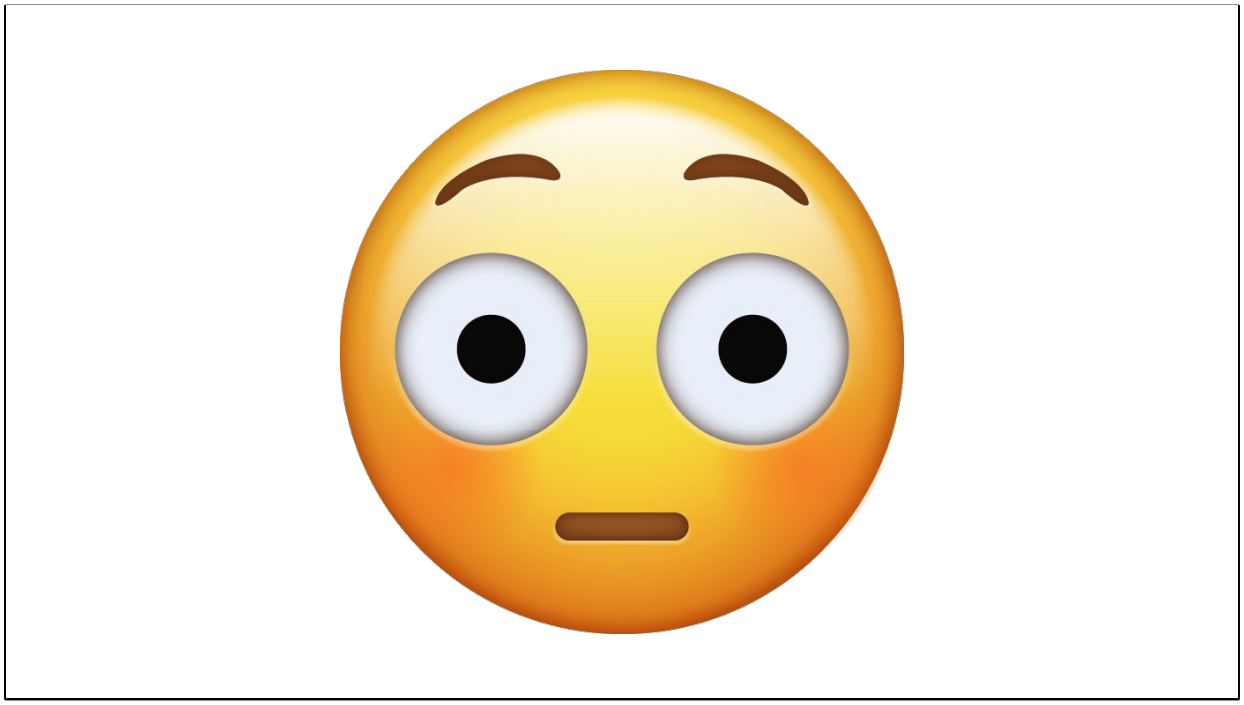


Not long ago, Adam Shostack posted on LinkedIn about a security warning he had received.



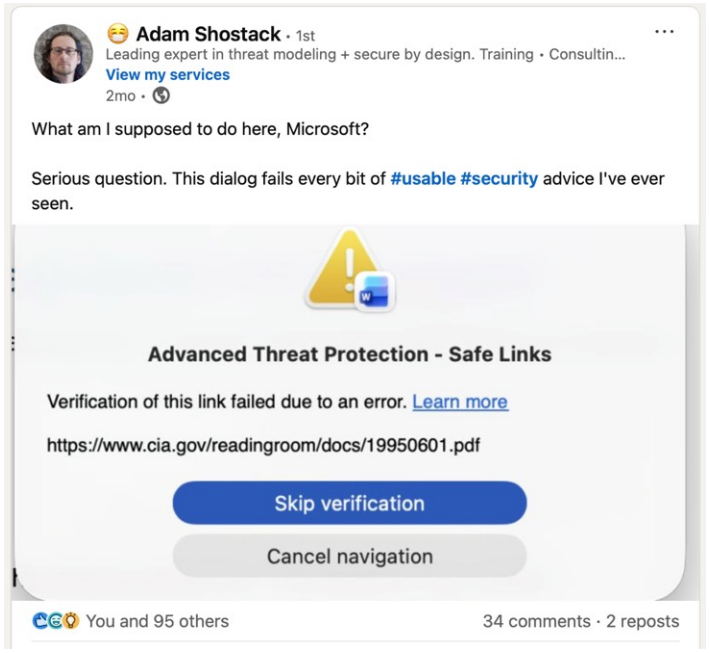
If you're like me, your reaction is something like...

Source/Image: https://www.linkedin.com/posts/shostack_usable-security-activity-7416636096668774400-qtOU/



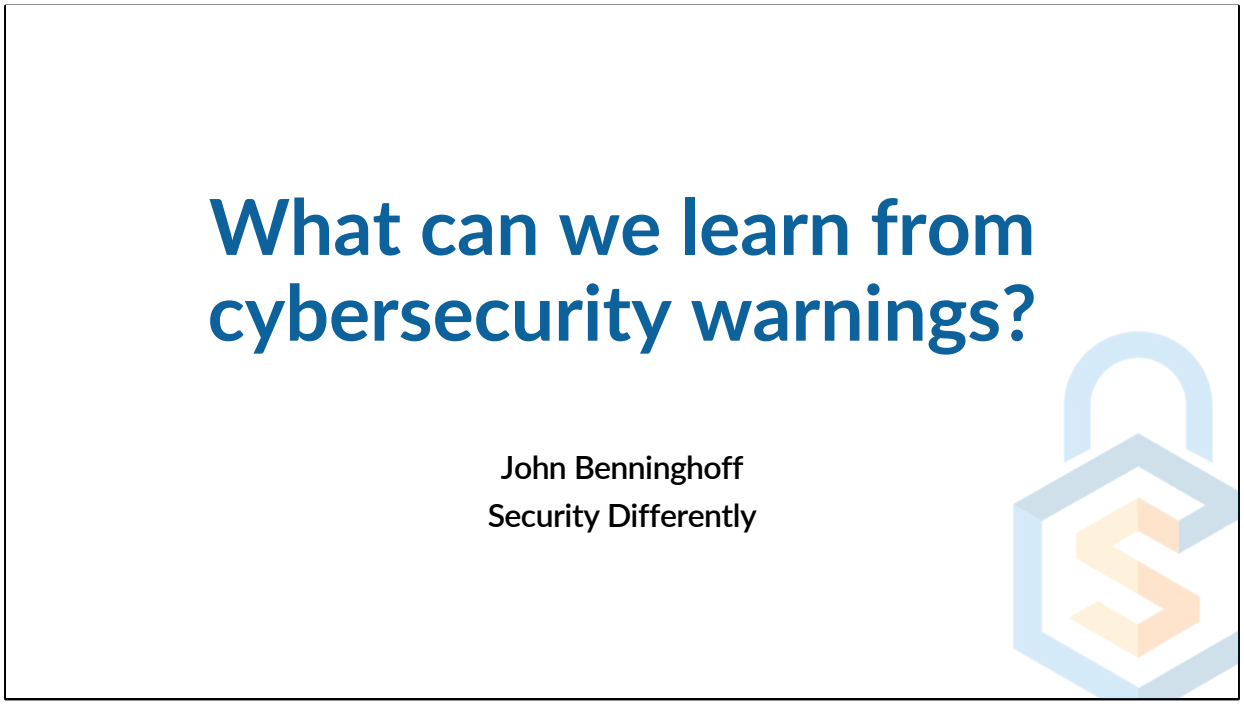
Wat?

Image: https://emojiisland.com/products/flushed-iphone-emoji-jpg



The irony here is that Adam was a coauthor of a Microsoft paper on how to write good security warnings that was published nearly 15 years ago (2012).

Source/Image: https://www.linkedin.com/posts/shostack_usable-security-activity-7416636096668774400-qtOU/



This is the result of an ongoing collaboration with a former coworker who led our UX team. Her PhD thesis was on safety warning labels.

I'll have a QR code at the end for you to download the slides with notes and links to all the references.

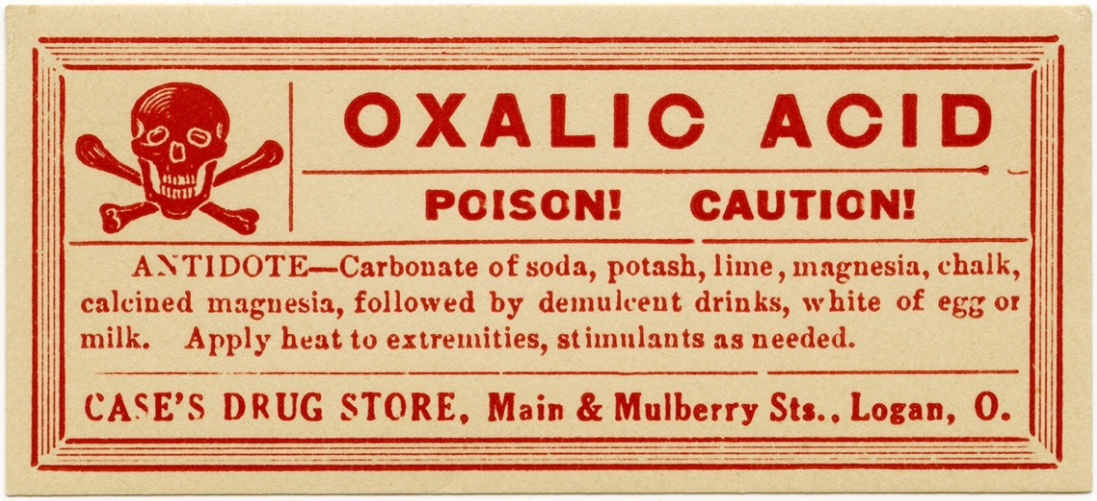
When I was working on this presentation, I thought of the 90s TV show Sliders. In it, a group travels to parallel universes through a wormhole. The worlds of safety and cybersecurity are a bit like parallel universes, with different histories, norms, and rules. Sometimes I feel like a slider, trying to explain how the world might be different.

https://en.wikipedia.org/wiki/Sliders_(TV_series)

A brief history

(of safety and cybersecurity warnings)

Let's briefly cover the history of the safety world and the cybersecurity world.



In the Safety World, factory worker deaths and the consumer safety movement in the early 20th century led to safety laws that included warning requirements. "In 1927, the United States enacted its first federal warning legislation, the Federal Caustic Poison Act (FCPA)."

Source: Wogalter, M. S. (2006). *Handbook of warnings*. Lawrence Erlbaum Associates. https://www.routledge.com/Handbook-of-Warnings/Wogalter/p/book/9780805847246
Image: https://olddesignshop.com/2014/10/oxalic-acid-poison-label-free-vintage-clip-art/

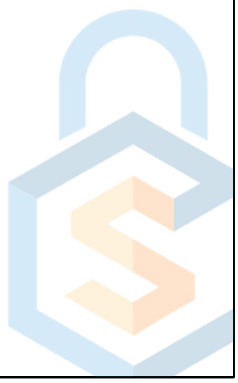
Industrial warnings were first standardized in 1941 with publication of ASA/ANSI Z35.1, which was succeeded by ANSI Z535. (image from Z35-1968). (The radiation symbol was added in 1959 and biohazard in 1968.

Source: https://en.wikipedia.org/wiki/ANSI_Z35
Image: https://commons.wikimedia.org/wiki/File:Z35-1968_Sign_-_Danger_-_High_Voltage.svg
Bonus:
https://web.archive.org/web/20120213165520/http://www.hms.harvard.edu/orsp/coms/biosafetyresources/history-of-biohazard-symbol.htm

Importantly, safety warnings have been driven by employer and product liability that established a **duty to warn.**

Source: Wogalter, M. S. (2006). *Handbook of warnings*. Lawrence Erlbaum Associates. https://www.routledge.com/Handbook-of-Warnings/Wogalter/p/book/9780805847246
Image: https://commons.wikimedia.org/wiki/File:16_CFR_§_1205.6_Warning_label_for_reel-type_and_rotary_power_mowers.svg

What about cybersecurity?

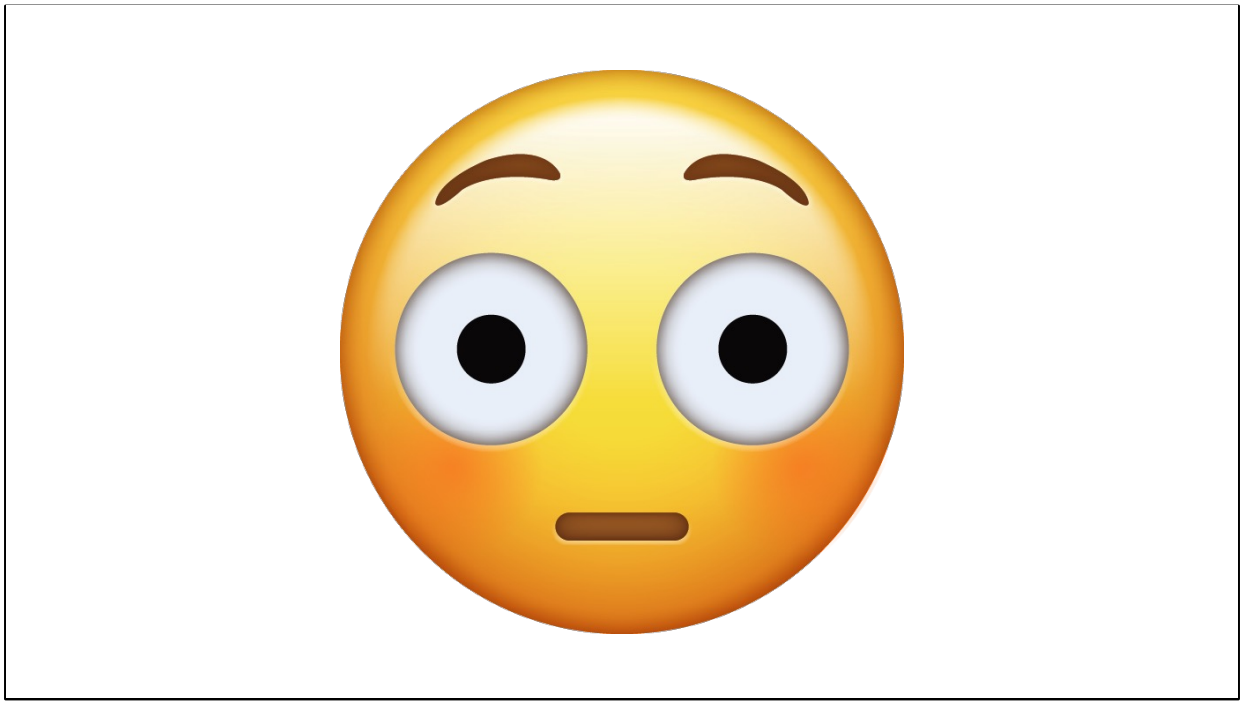


The world of Security Software didn't even exist until 1976-1977, with the creation of RACF and ACF2, respectively. Antivirus software, an early source of warnings, started in the 1980s-1990s. One problem: early antivirus software would sometime flag a benign file as a virus (false positive).

Figure 3-5. System Scan response options

McAfee VirusScan, 1999. Clean = try to clean, Continue = ignore and keep scanning, Stop = stop scanning, Exclude = mark as safe and continue, Delete = only safe option.

Image/Source: http://archive.org/details/mcafee-virusscan-for-windows-95-and-windows-98-users-guide-version-4.0.1



Wat?

Image: https://emojiisland.com/products/flushed-iphone-emoji-jpg

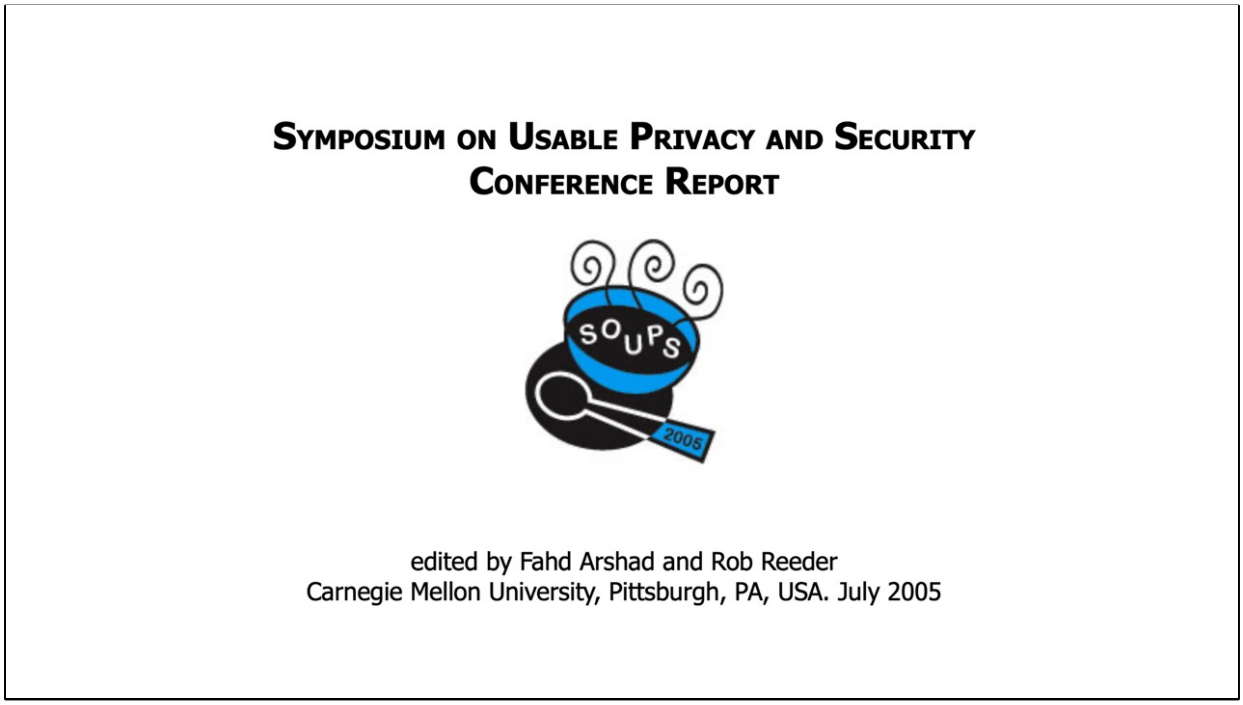
Why Johnny Can't Encrypt:
A Usability Evaluation of PGP 5.0

Alma Whitten
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
alma@cs.cmu.edu

J. D. Tygar[1]
EECS and SIMS
University of California
Berkeley, CA 94720
tygar@cs.berkeley.edu

Thankfully, things were about to change. The seminal 1999 paper, Why Johnny Can't Encrypt, reported on a lab test of PGP. Only 4 of the 12 participants were able to correctly sign and encrypt an email message in 90 minutes, in large part because the system design assumed at least working technical knowledge of how public key encryption worked.

Source: https://www.usenix.org/conference/8th-usenix-security-symposium/why-johnny-cant-encrypt-usability-evaluation-pgp-50, https://www.usenix.org/legacy/events/sec99/full_papers/whitten/whitten.pdf



A few years later, in 2005, the first SOUPS conference was held at Carnegie Mellon, with a focus on security usability, including better security warnings. Early SOUPS conferences benefited from the Microsoft Trustworthy Computing Initiative, launched in Jan 2002 (and ended in 2014). SOUPS has been held every year since and is now run by USENIX.

Source: https://cups.cs.cmu.edu/soups/, https://www.usenix.org/conferences/byname/884, https://web.archive.org/web/20150626152000/https://threatpost.com/era-ends-with-break-up-of-trustworthy-computing-group-at-microsoft/108404
Image: https://cups.cs.cmu.edu/soups/2005/SOUPS_2005_Conference_Report.html

NEAT Framework (SOUPS 2011)

Security warnings should be:

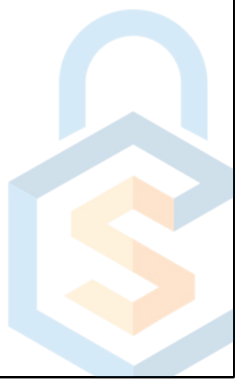
- Necessary
- Explained
- Actionable
- Tested

Rob Reeder, Ellen Cram Kowalczyk, and Adam Shostack developed the NEAT framework while at Microsoft. Presented at SOUPS 2011.

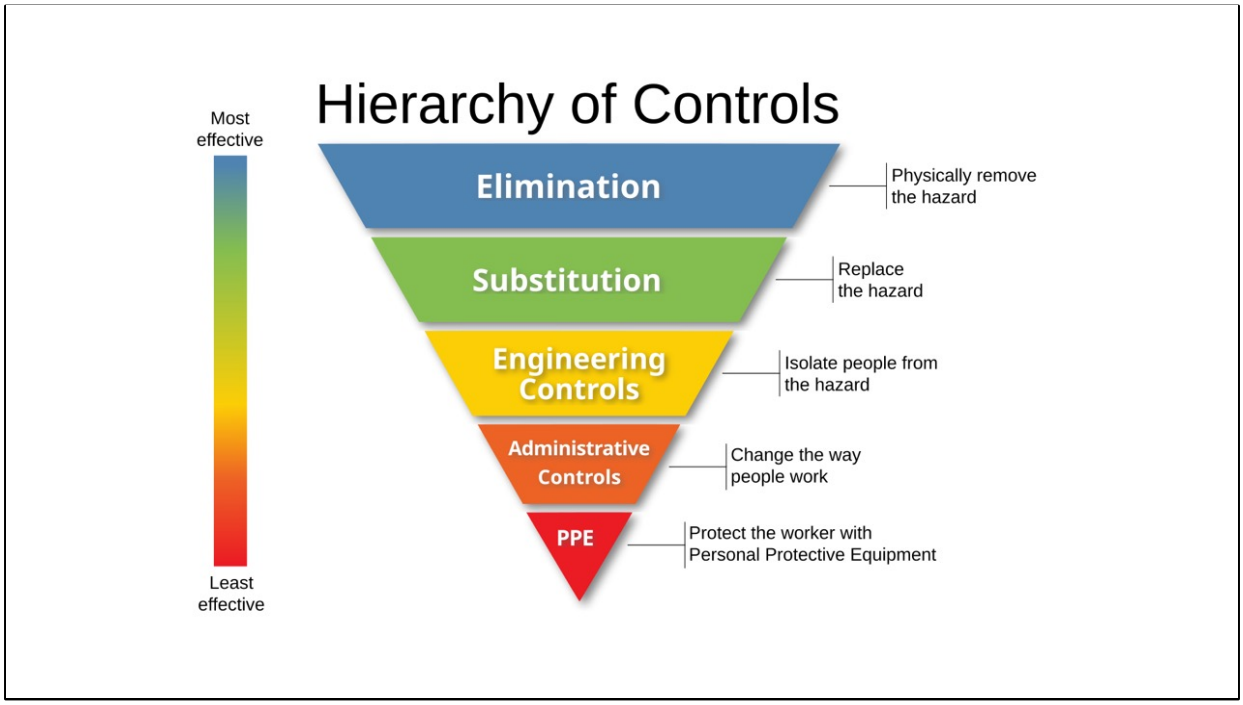
"**Necessary**: A warning should only interrupt a user if it is absolutely necessary to involve the user. Sometimes, a system can automatically take a safe course of action without interrupting the user. Sometimes, a security decision can be deferred to a later point in time.
Explained: If it is actually necessary to interrupt the user with a security warning, the warning should explain the decision the user needs to make and provide the user with all the information necessary to enable them to make a good decision. Since the Explained part of NEAT is perhaps the most important, we devised another acronym, CHARGE US (see below), to help engineers remember what information to provide in a security warning
Actionable: A security warning should only be presented to the user if there is a set of steps the user could realistically take to make the right decision in all scenarios, both benign (where there is no attack present) and malicious (where an attack is present).
Tested: Security warnings should be tested by all means available, including visual inspection by many eyes and formal usability testing."

Source: https://shostack.org/files/papers/ReederEtAl_NEATatMicrosoft.pdf, https://cups.cs.cmu.edu/soups/2011/program.html

Lessons from safety



Given that history, what can cybersecurity learn from the world of safety?



The Hierarchy of Hazard Controls is an idea from safety that's overdue to be adopted in technology. It's a reminder that warnings are the *least* effective control, and when a more effective control is feasible, it should be used instead (or in addition to). This can be simplified as: Design, Guarding, Warning.

Source: https://en.wikipedia.org/wiki/Hierarchy_of_hazard_controls, https://journals.sagepub.com/doi/10.1177/1557234X0600200109
Image: https://commons.wikimedia.org/wiki/File:NIOSH's_"Hierarchy_of_Controls_infographic"_as_SVG.svg

C-HIP Model

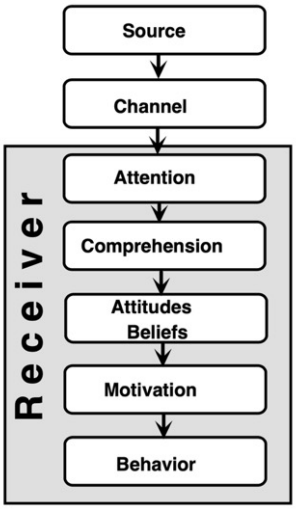
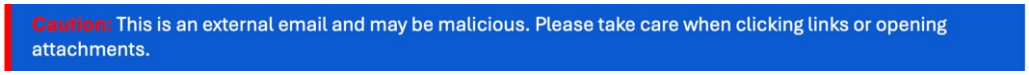


Fig. 2. Communication-human information processing (C-HIP) model. (Wogalter et al., 1999b).

The Communications-Human Information Processing (C-HIP) model provides a framework for evaluating the effectiveness of warnings, where the goal is to change behavior. The warning can fail at any stage in processing the message.

Source: https://journals.sagepub.com/doi/10.1177/1557234X0600200109
Image: https://www.researchgate.net/publication/11221385_Research-Based_Guidelines_for_Warning_Design_and_Evaluation

Adaption



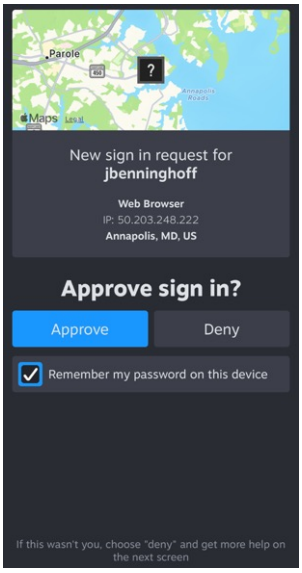
One problem with warnings is adaption; repeated exposure to the same warning leads to ”inattentional blindness” – we ignore the warning since we’ve seen it so many times. Phishing Banners are an example. How many have “external” banners at work? Do you still notice them? Does it affect your behavior?

This also fails to account for the Hierarchy of Controls - phishing is better handled through guarding (blocking malicious emails) and design (phishing-resistant MFA = passkeys).

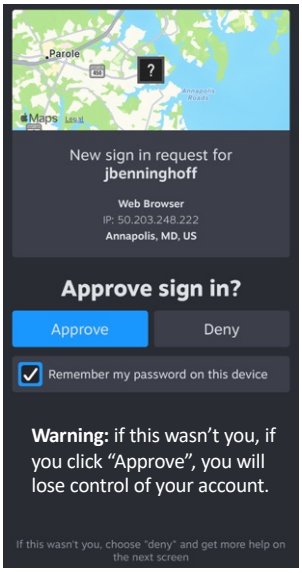
Explicitness

“Explicitness in this context is defined as information that is specific, detailed, clearly stated, and leaves nothing implied.” - “• Do not assume “everybody knows.” • Do not rely on inference. • Be careful about assuming that hazards and consequences are open and obvious. • People do not always remember the appropriate safety information at the appropriate time. Reminders may be needed. • **Explicit is not necessarily synonymous with quantitative.** • Technical jargon is usually not a good way to achieve explicitness, especially for a general target audience.”

Source: https://journals.sagepub.com/doi/10.1177/1557234X0600200109
Image: https://commons.wikimedia.org/wiki/File:ANSI_Z535_Style_sign_Warning_010.svg

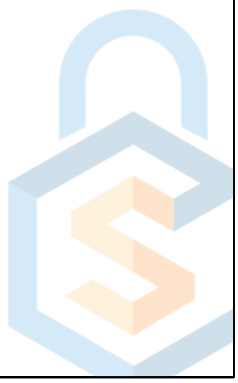


I took the picture on the left while hiking; the sign is quite explicit about the hazard and outcome. The screenshot on the right is a typical “approve sign-in screen."



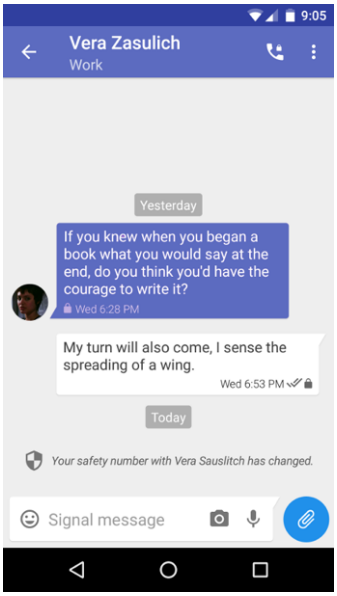
A better warning for the approve sign-on screen on the right!

Lessons from research



What can we learn from cybersecurity usability research?

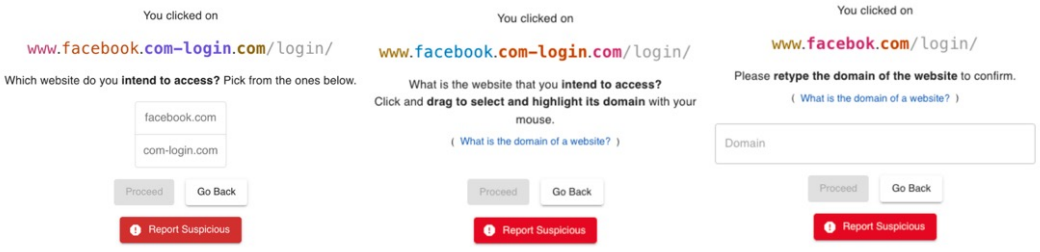
Consider the System



Title of paper: "Something isn't secure, but I'm not sure how that translates into a problem": Promoting autonomy by designing for understanding in Signal. The “safety number has changed” warning violates the explicitness principle, and doesn’t account for the fact that “Users’ strategies for coping with online threats extend beyond the ecosystem of your app.” – if you tell people instead that Vera most likely got a new phone, but there’s a small chance that your messages are now being intercepted, they can adjust their conversations (avoiding sensitive discussion) or take action outside the Signal app.

Source: https://www.usenix.org/conference/soups2019/presentation/wu
Image: https://signal.org/blog/verified-safety-number-updates/

Slow thinking beats fast thinking



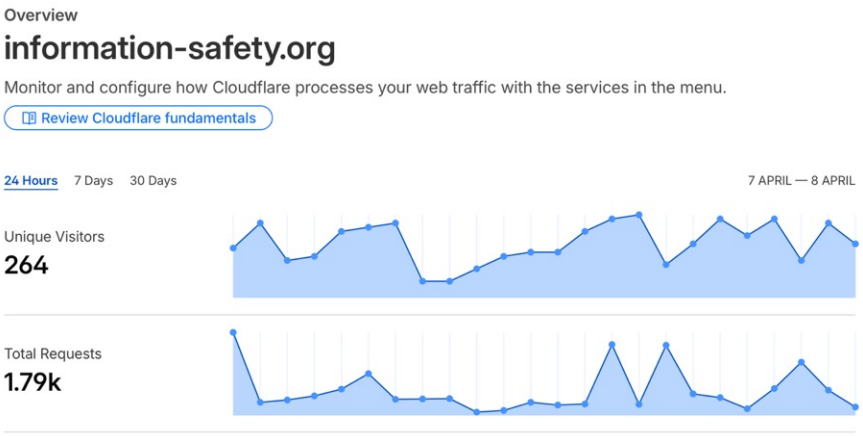
(a) Clicking Task. (b) Highlighting Task. (c) Typing Task.

Figure 2: Three selected tasks for active URL inspection after clicking on a link.

Title: “URL Inspection Tasks: Helping Users Detect Phishing Links in Emails”. While the premise of the paper is completely wrong (implement phishing-resistant MFA instead of warning the user), it does demonstrate a useful fact: turning URL recognition from a fast-thinking task into a slow-thinking task improved user performance. If you must warn, invoking slow thinking promotes success in decision-making.

Source, Image:
https://www.usenix.org/conference/usenixsecurity25/presentation/lain

Measurement

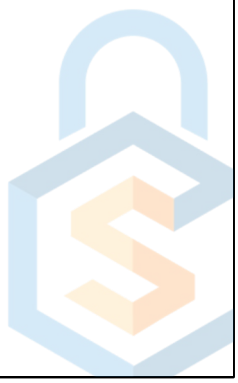


One of the challenges of traditional warnings is measuring their effectiveness, which is costly. Software systems can instrument and monitor nearly anything and everything, making measurement easier. (the T in NEAT). Google ‘unsafe link’ example.

Source, Image: Cloudflare Dashboard

Cybersecurity Warnings

(examples from software systems)



Let's look at some examples of cybersecurity warnings.

Types of cybersecurity warnings

- **Action-Required**
- Judgement-Required

Key insight: there are two types of cybersecurity warnings:

Action Required: Hazard is known, but the user must act because the system can't.
Judgement Required: Hazard may or may not be present, and the user must make a risk decision.

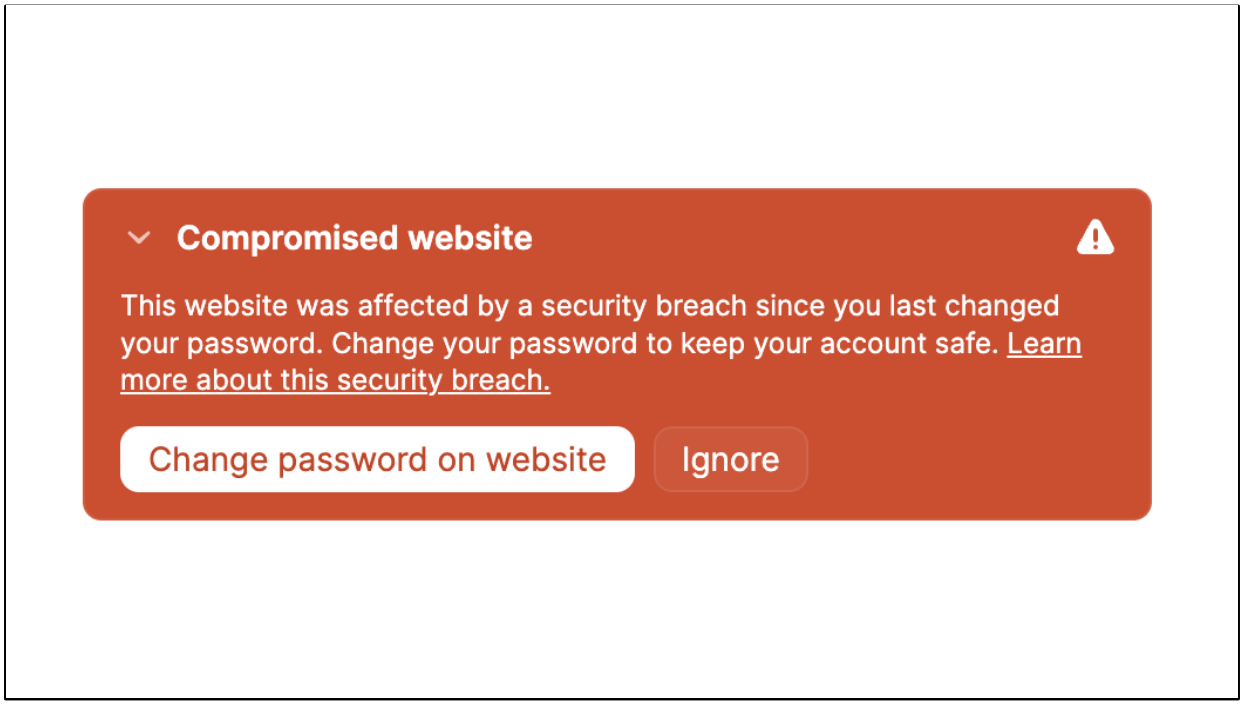
We'll look at action-required first (less common).

Revised NEAT Framework

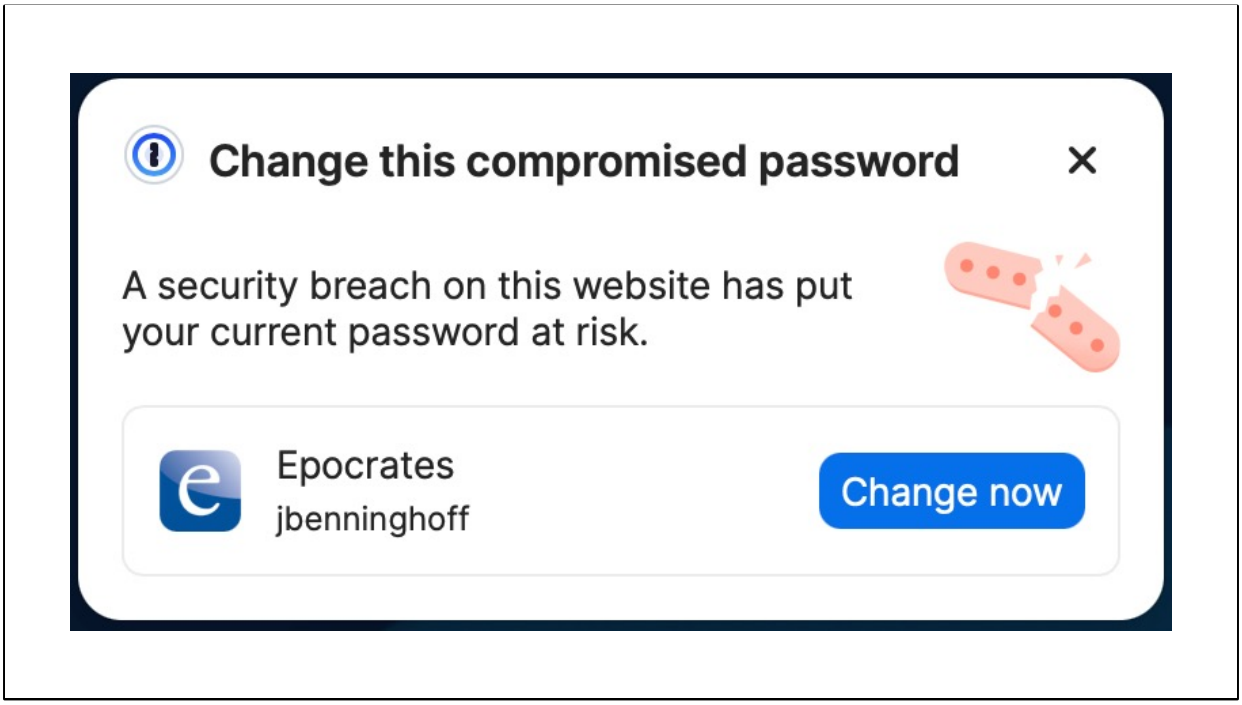
Security warnings should be:

- Necessary (can't design or guard out)
- Explicit
- Actionable (or decision required)
- Tested

To evaluate these warnings, I'll use an updated version of NEAT, that more directly includes lessons from safety.



Action-required example: in the 1Password App, selecting a Login that has experienced a breach generates a warning banner above the Login entry. NEAT!

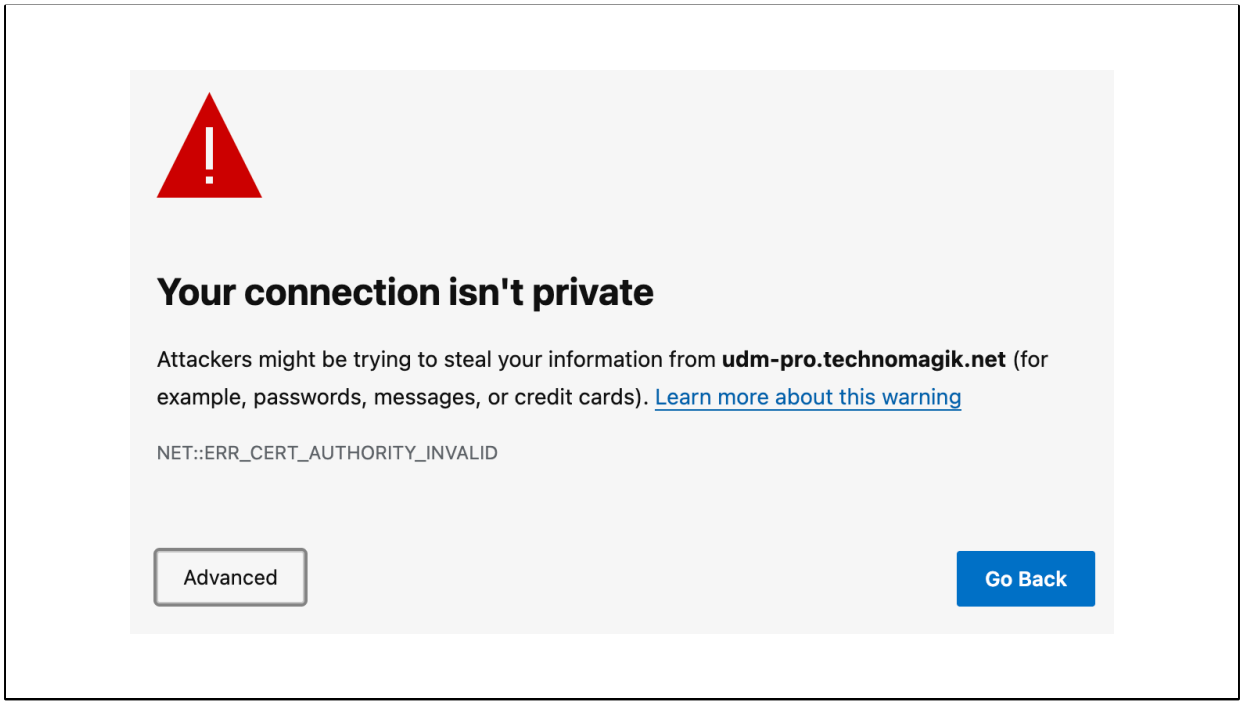


If the banner warning is ignored, 1Password displays an additional warning after autofilling the username and password, prompting a password change when it's most accessible. (Good design, NEAT)

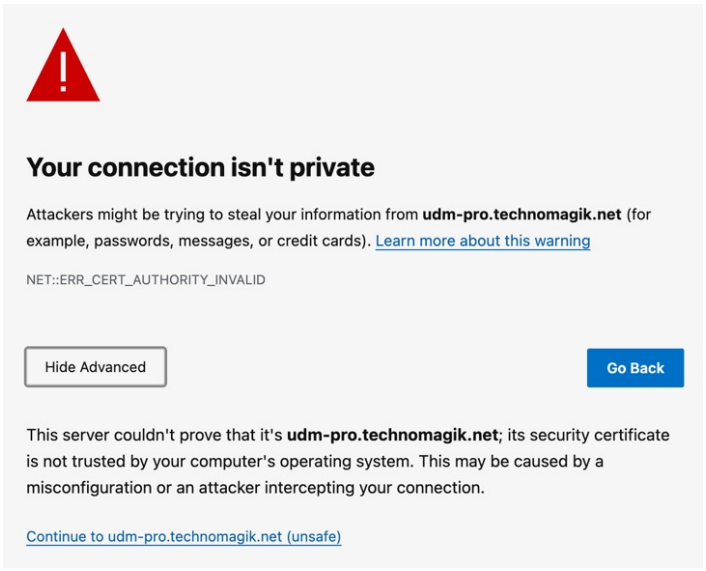
Types of warnings

- Action-Required
- **Judgement-Required**

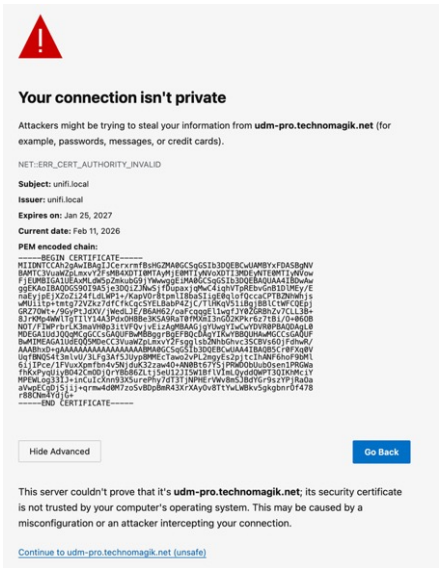
Next, we have an example of judgement-required.



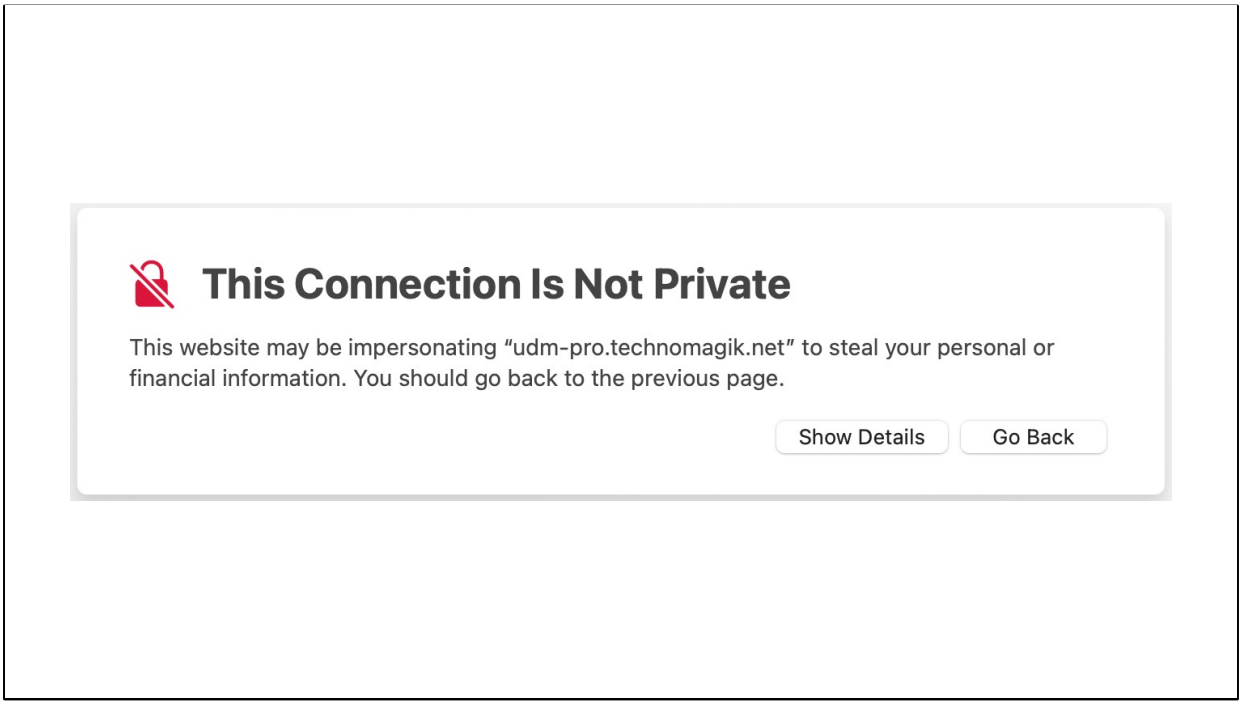
Microsoft Edge certificate warning. Judgement required, is this NEAT? It uses some gating (must click Advanced to continue) = N.



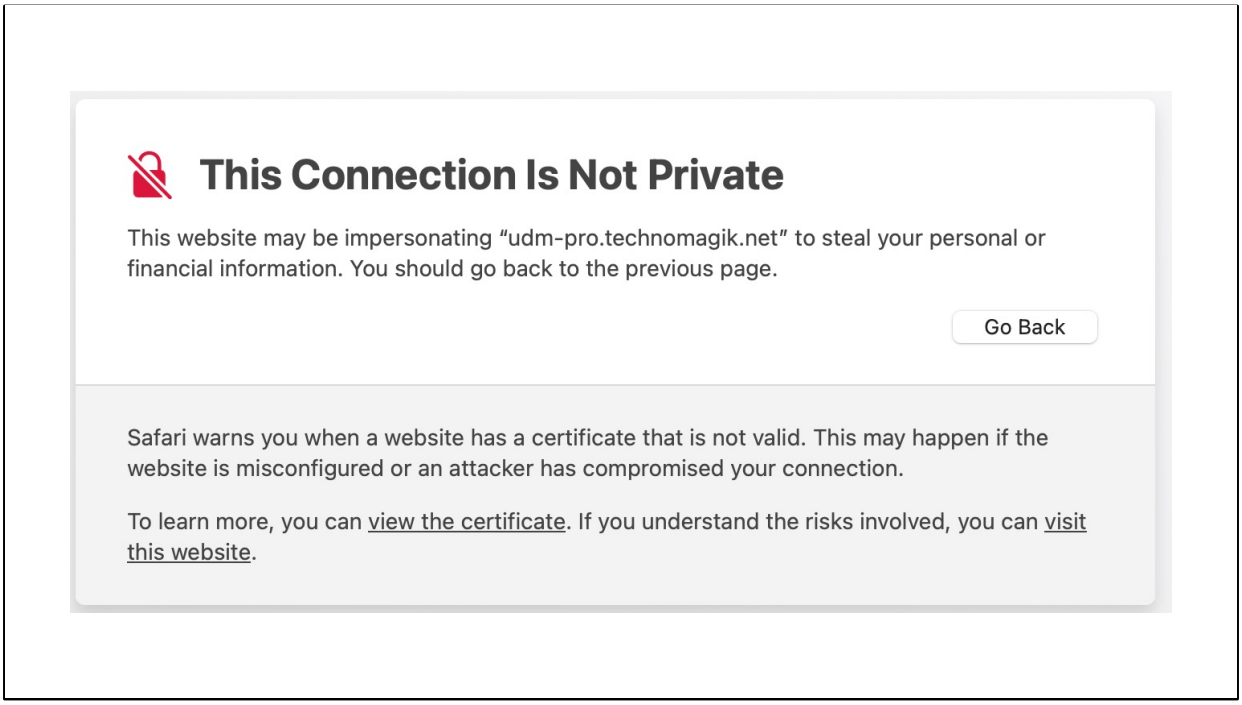
This is somewhat explicit but uses jargon and doesn't describe the most common scenarios (self-signed or expired cert vs small chance of listening in). Also, where's the certificate? I might need that information to decide..



Clicking on “NET:ERR_CERT_AUTHORITY_INVALID” reveals some certificate information. Why is this hidden?

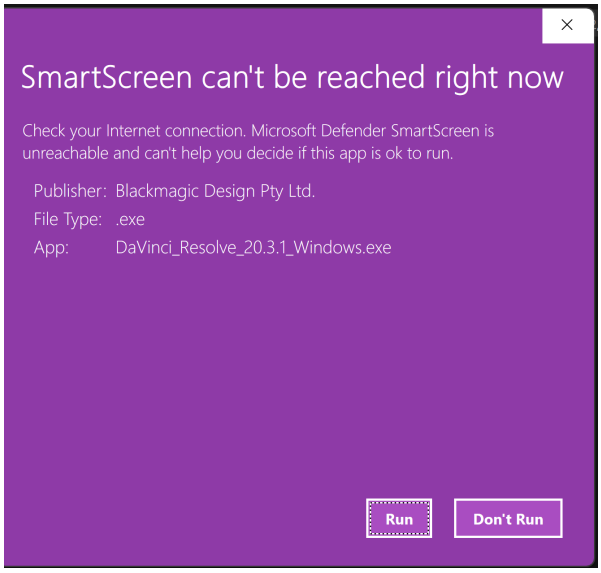


Safari certificate warning. Is this NEAT? Also gated (N)

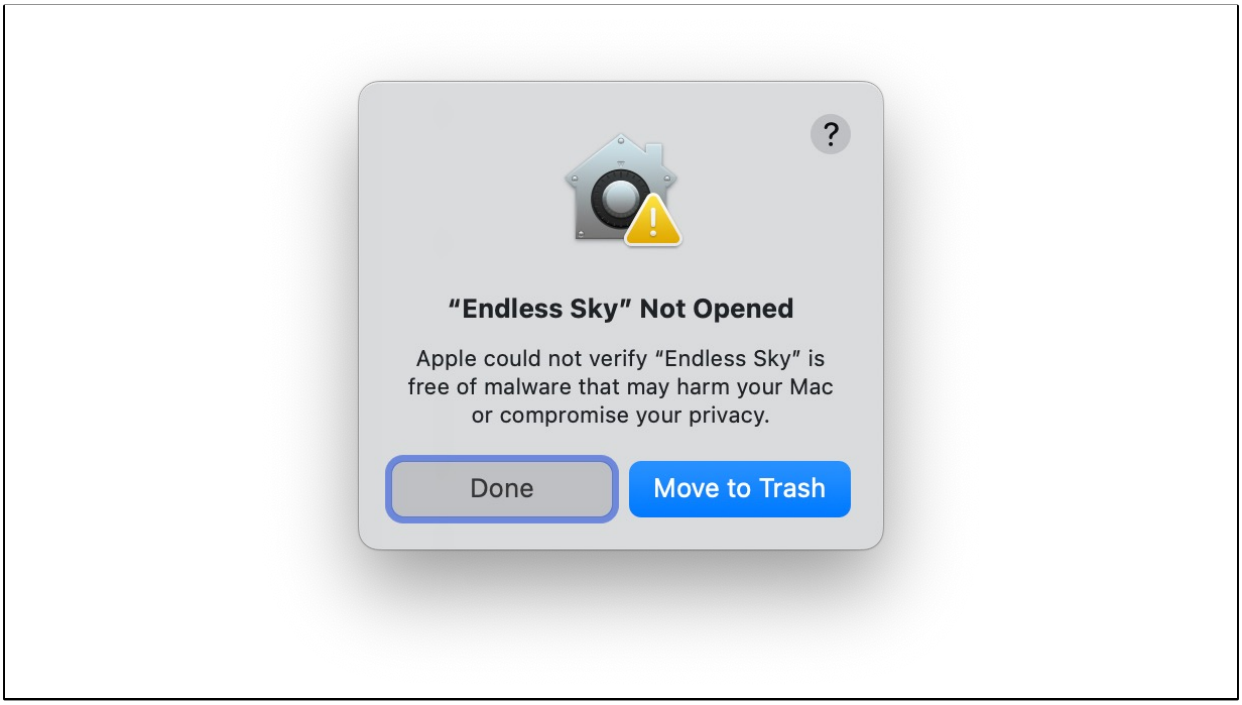


A bit more explicit, but the language could be cleaner and include common examples (self-signed certificate).

Safari shows you the entire certificate, which is better; you can see that the cert is self-signed.



Is this NEAT? Could be solved through improved design (probably not necessary).

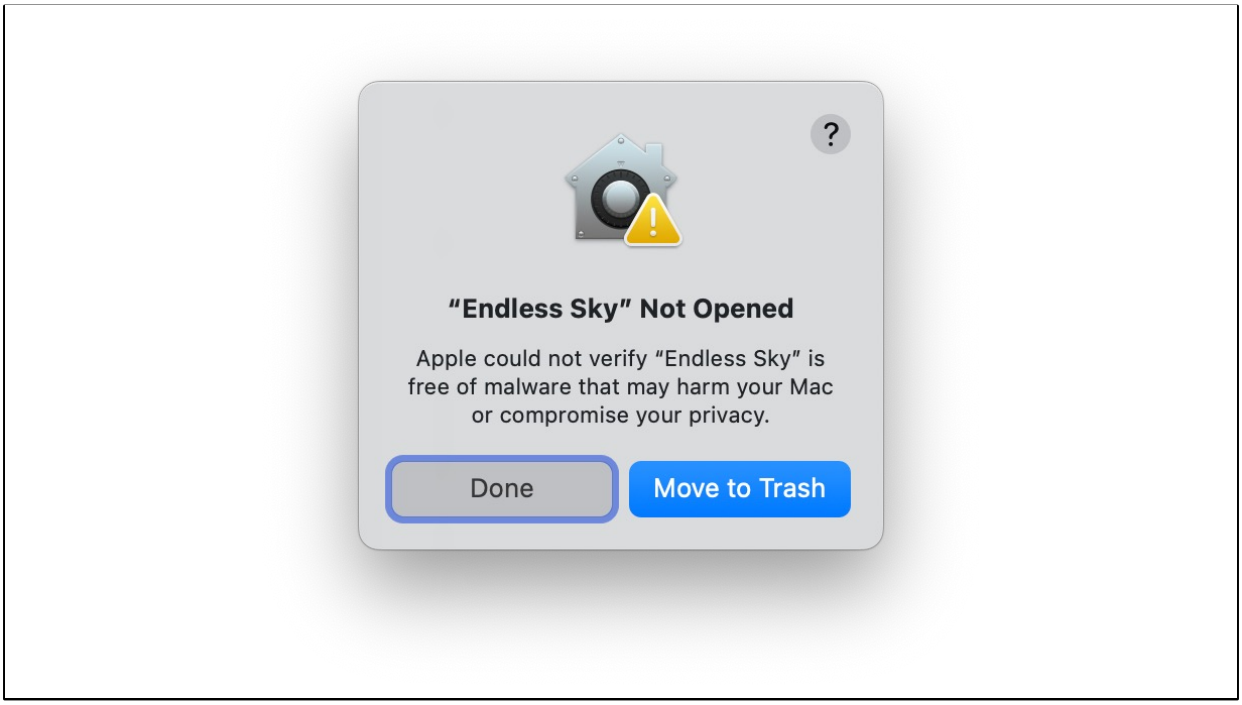


This is one of the most sophisticated security warnings I've found! This warning appears when opening an unsigned application on macOS.

I bet you're thinking something like this...

I bet you’re thinking something like this…

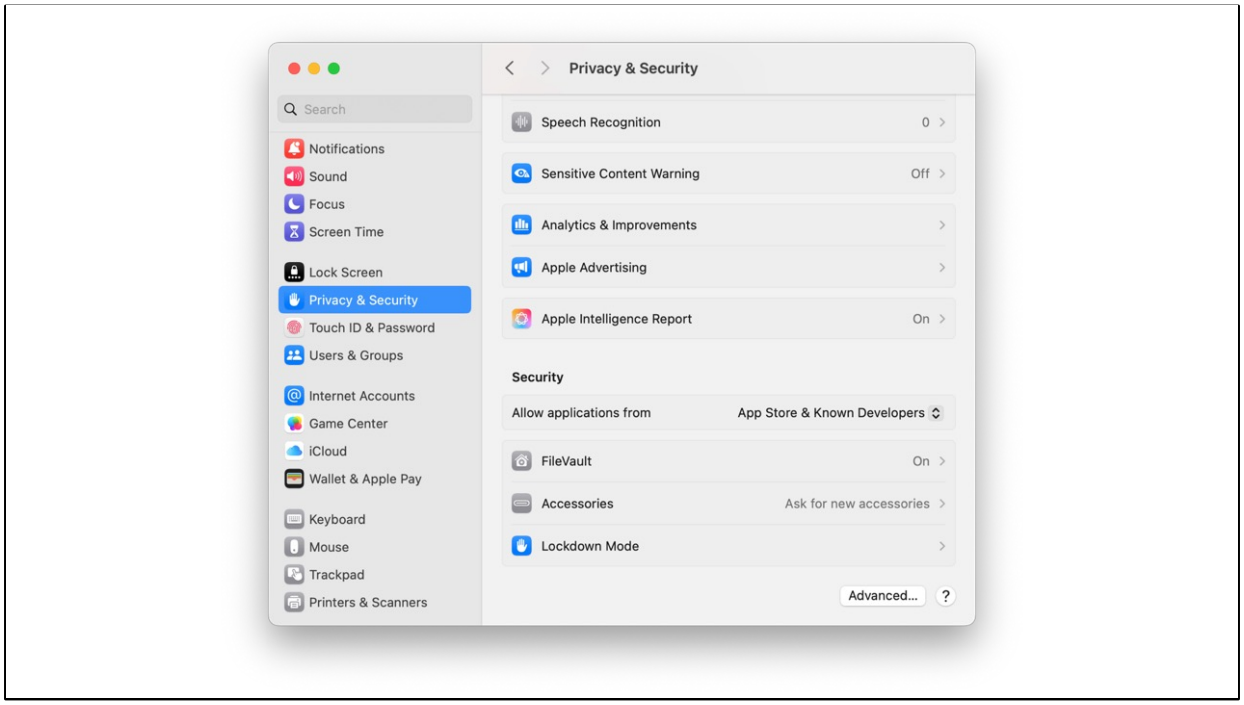
Image: https://emojiisland.com/products/unamused-iphone-emoji-jpg



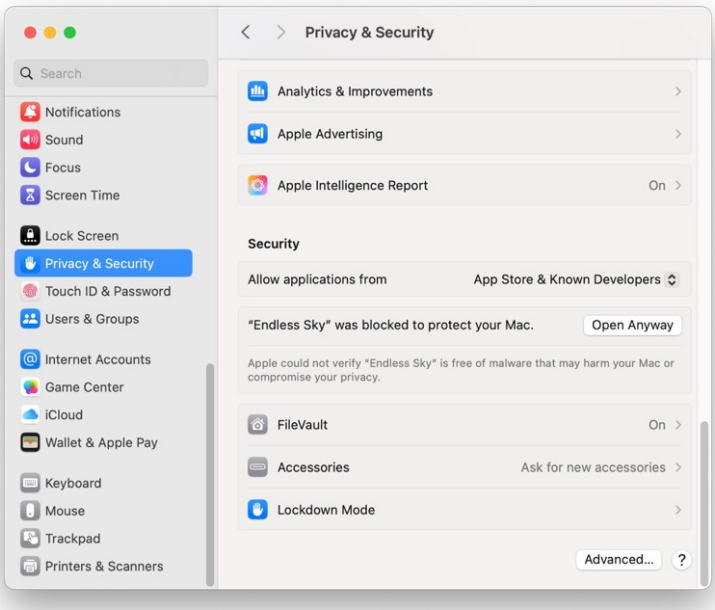
What makes this warning sophisticated? It's Necessary, Explicit, and only offers safe choices. But what if you understand the risks and want to run an unsigned application anyway?

IYKYK!

There's hidden gating here – if you know how macOS security works, you can bypass the warning. This is a way of gating based on the user's knowledge, to see if they are "qualified" to make an informed decision to ignore the warning.



This is what the Privacy & Security tab in System Settings normally looks like.

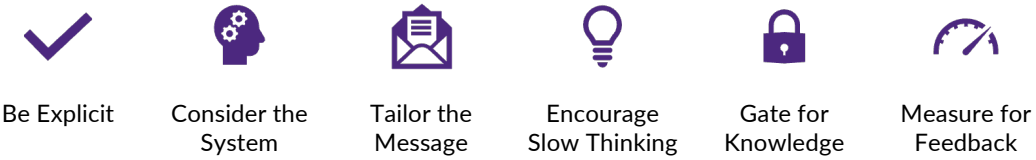


After you try to open an unsigned application, an extra setting will appear here – if you know it's there, you can bypass the gating and open the application.

Lessons for Risk Communication & Decisions

What does this all mean for risk communications and decisions?

Lessons for Risk Communication & Decisions



I think these lessons apply broadly to risk communication. Many of these we already do today.

Ignoring the warning is not a failure

If the person makes a well-informed decision to ignore it.

And...

Thank you!

Slides, Connect & Website

Slides:
jbenninghoff.com/qr

Connect:
linkedin.com/in/jbenninghoff/

Website:
jbenninghoff.com
security-differently.com

Scan the QR code for slides and more!