16- SSL, PKIs, and Secure Communication

Lujo Bauer, Nicolas Christin, and Abby Marsh

March 14, 2016

05-436 / 05-836 / 08-534 / 08-734 / 19-534 / 19-734
Usable Privacy and Security

Carnegie Mellon University
CyLab

institute for SOFTWARE RESEARCH

Engineering & Public Policy

Today!

- An introduction to SSL/TLS
- An introduction to PKIs
- Recent developments in this area
- Usability issues
- An activity to make it better

Overview

- Secure Sockets Layer (SSL) and its successor, Transport Layer Security (TLS) enable secure communication
- Frequently encountered with web browsing (HTTPS) and more behind the scenes in app, VOIP, etc.

What we want to defend against

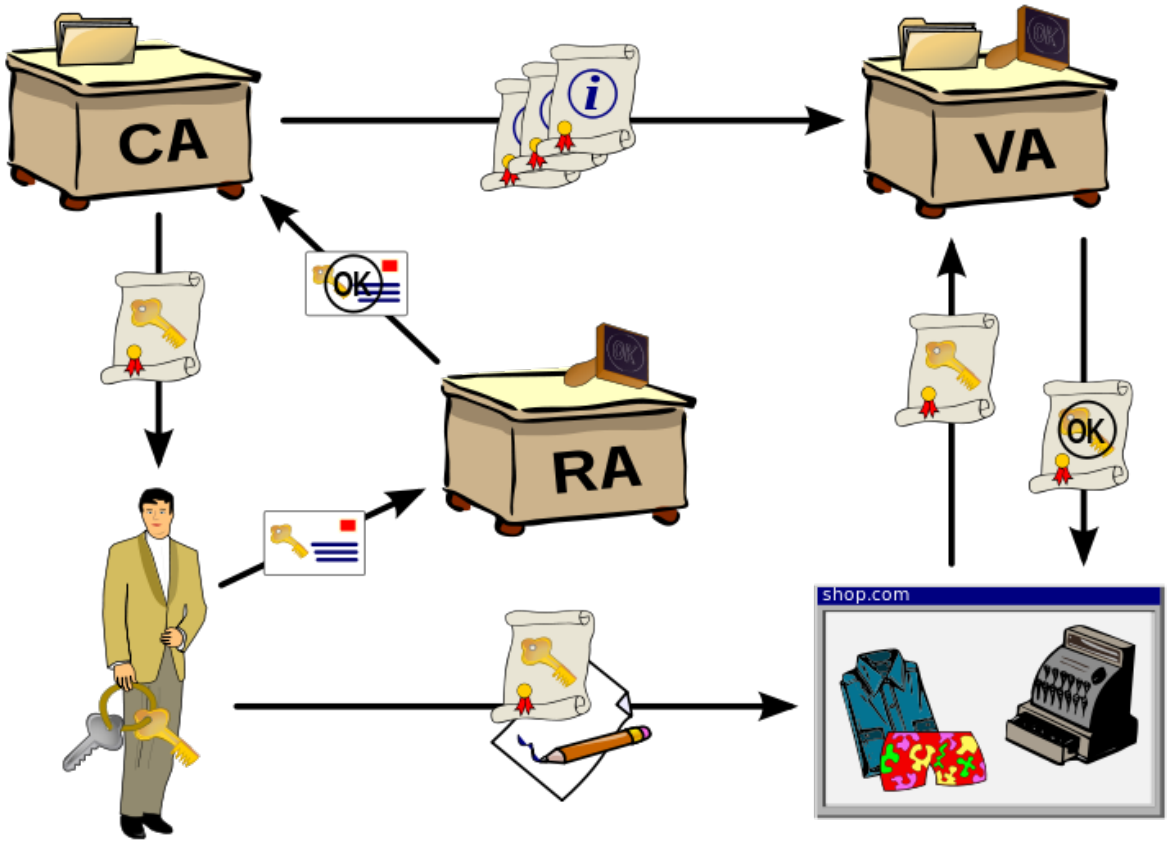
- People snooping on our communications
  - The contents of what we're sending
  - Session tokens (see, e.g., Firesheep)
- Man-in-the-middle attacks
  - We want to authenticate that we are talking to the right site, not an imposter
  - Use certificates inside a public-key infrastructure

How we could obtain trust

- Web of trust
  - People you already trust introduce you to people they trust
  - Can get complicated, doesn't scale well
  - Less frequently seen in practice
- Public-Key Infrastructure (PKI)
  - Certificates are issued by certificate authorities that bind cryptographic keys to identities

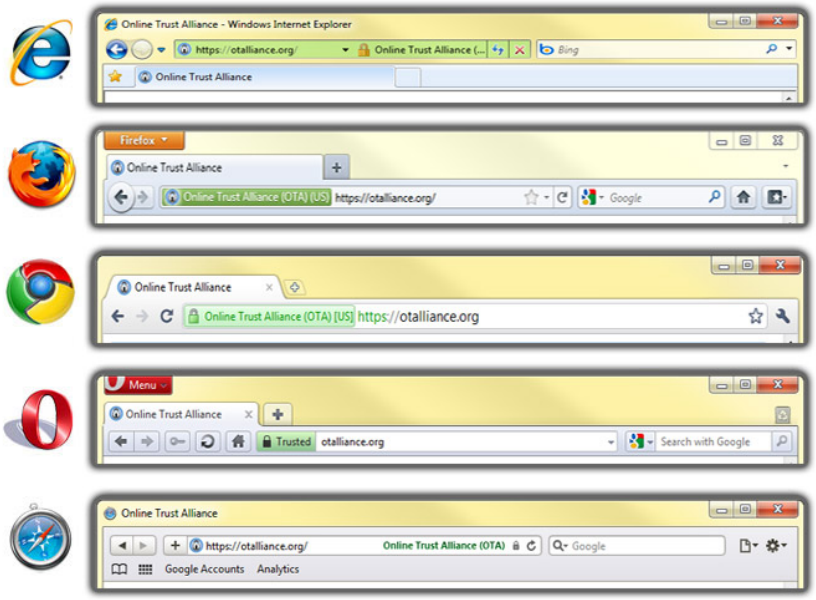
Public-Key Infrastucture

- Binding of keys to identities can be done automatically or by humans



What does SSL look like to users?

- Compare, e.g., the following:
  - https://www.google.com (normal certificate)
  - Go to Google images and then click on an image and see what happens (mixed content)
  - https://otalliance.org (EV certificate)



What does PKI look like to browsers?

- Hundreds of trusted certificate authorities
  - Certificate authorities (CAs) sign the certificates binding identities to keys
  - See, e.g., Firefox's advanced settings

What does PKI look like to sites?

- Apply for a certificate
  - Validation process
  - Certificate authorities (CAs) delegate trust (“chain of trust”)
  - CAs sell you a certificate

Issues with SSL/TLS/PKIs

- Implementation issues
- Communicating to users what is happening
- Compromised Certificate Authorities
- Man-in-the-middle attacks
  - Downgrade/dumbing-down attacks
  - Addition of “rogue” certificates
- Revocation
- Timing attacks and other side channels

One famous implementation issue

- OpenSSL bug
  - Heartbleed (CVE-2014-0160)
  - TLS heartbeat extension misses a bounds check and thus lets an attacker “read” memory

Frequent implementation issues

- Shared code or forum posts can tell you how to “stop” errors about certificate validation
  - Fahl et al. CCS ‘13
- How do we help app developers write secure software?

Compromised CAs

- Comodo and Diginotar both suffered breaches in 2011 that let attackers issue rogue certificates
- What about untrustworthy CAs?
  - Compelled certificate creation attacks (see, e.g., Soghoian and Stamm FC '11)

Man-in-the-middle attacks (MITM)

- Effectively, many corporations perform MITM attacks by adding certificates to users' computers and presenting "fake" certificates to users.
- A man in the middle can also tell you a site doesn't support SSL/TLS (downgrade) or any strong ciphers (dumbing down)
  - Why does this create a huge problem?
  - Why is this hard to deal with?

Important question 1

- How do you know if a site supports HTTPS?
  - EFF's HTTPS Everywhere
  - HTTP Strict Transport Security (HSTS)
  - In both cases, how do you bootstrap/maintain?

Important question 2

- How do you know you have the right certificate for a site?
  - Certificate transparency
  - Public key pinning
  - Perspectives (originally a CMU project)

How do you know a cert is valid?

- Certificates can be revoked in case of a compromise
- Certificate Revocation Lists (CRLs) were used, but they got really large
  - Incremental updates were better
- Online Certificate Status Protocol (OCSP)
  - How does this impact privacy?
- OCSP Stapling

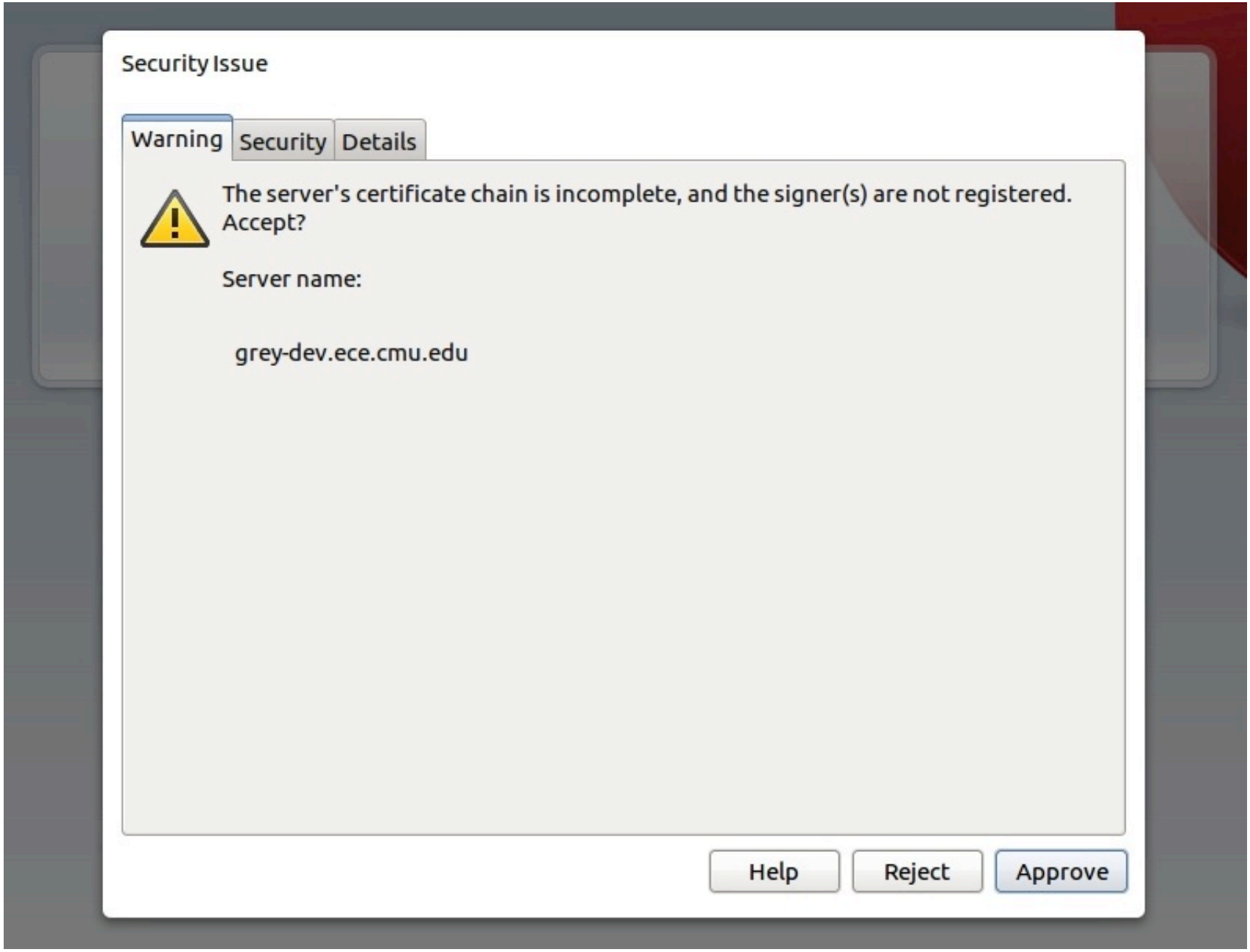
Self-signed certificates

- What happens if someone signs their own certificate and chooses not to use the PKI infrastructure?
  - You get a warning!

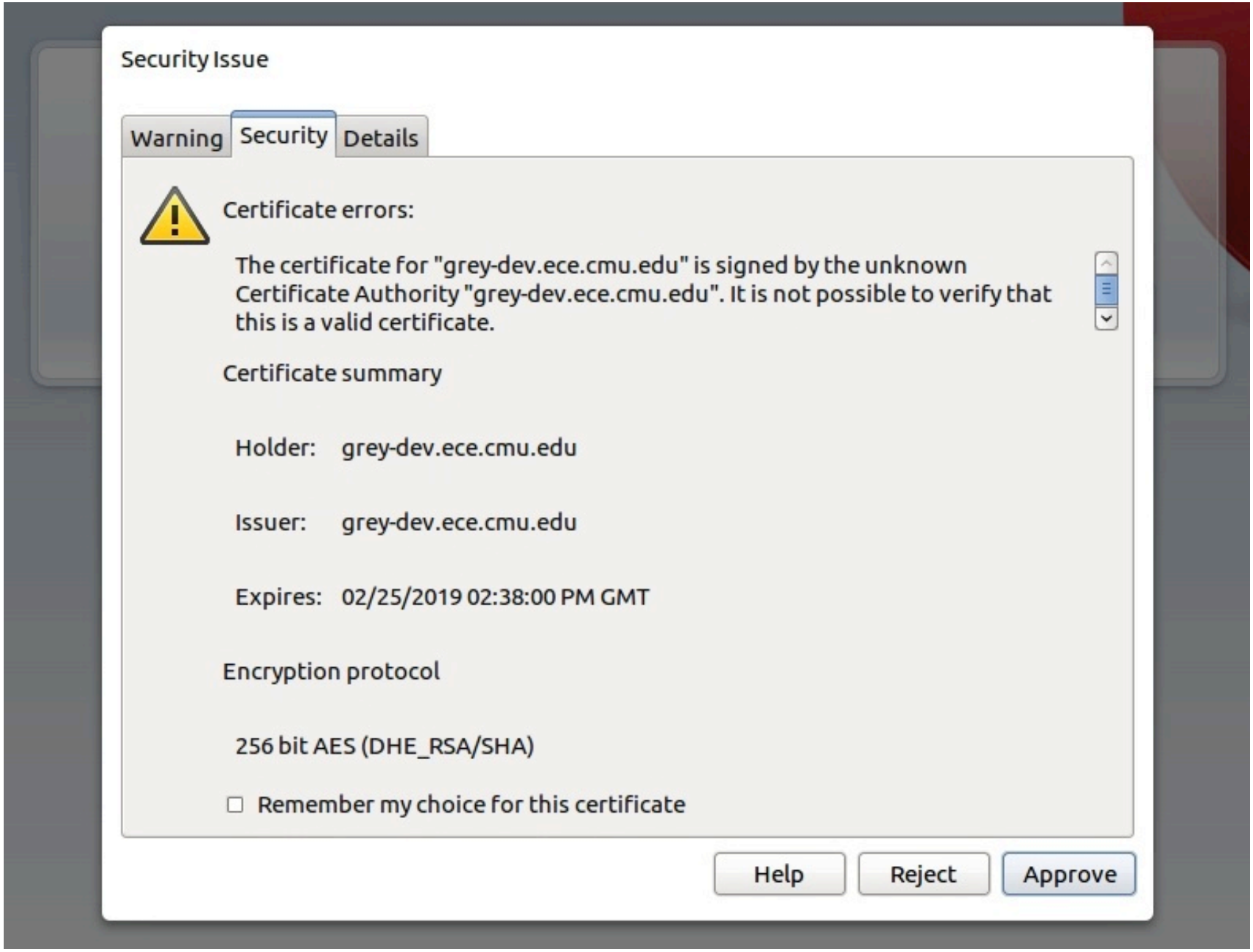
Warnings


http://www.utechsoft.com
This applet was signed by "Unlimi-Tech Software Inc.," and authenticated by "Thawte Consulting cc". Do you trust this certificate?
Click Trust to run this applet and allow it unrestricted access to your computer. Click Don't Trust to run this applet with standard Java restrictions.
?
Show Certificate
Don't Trust
Trust

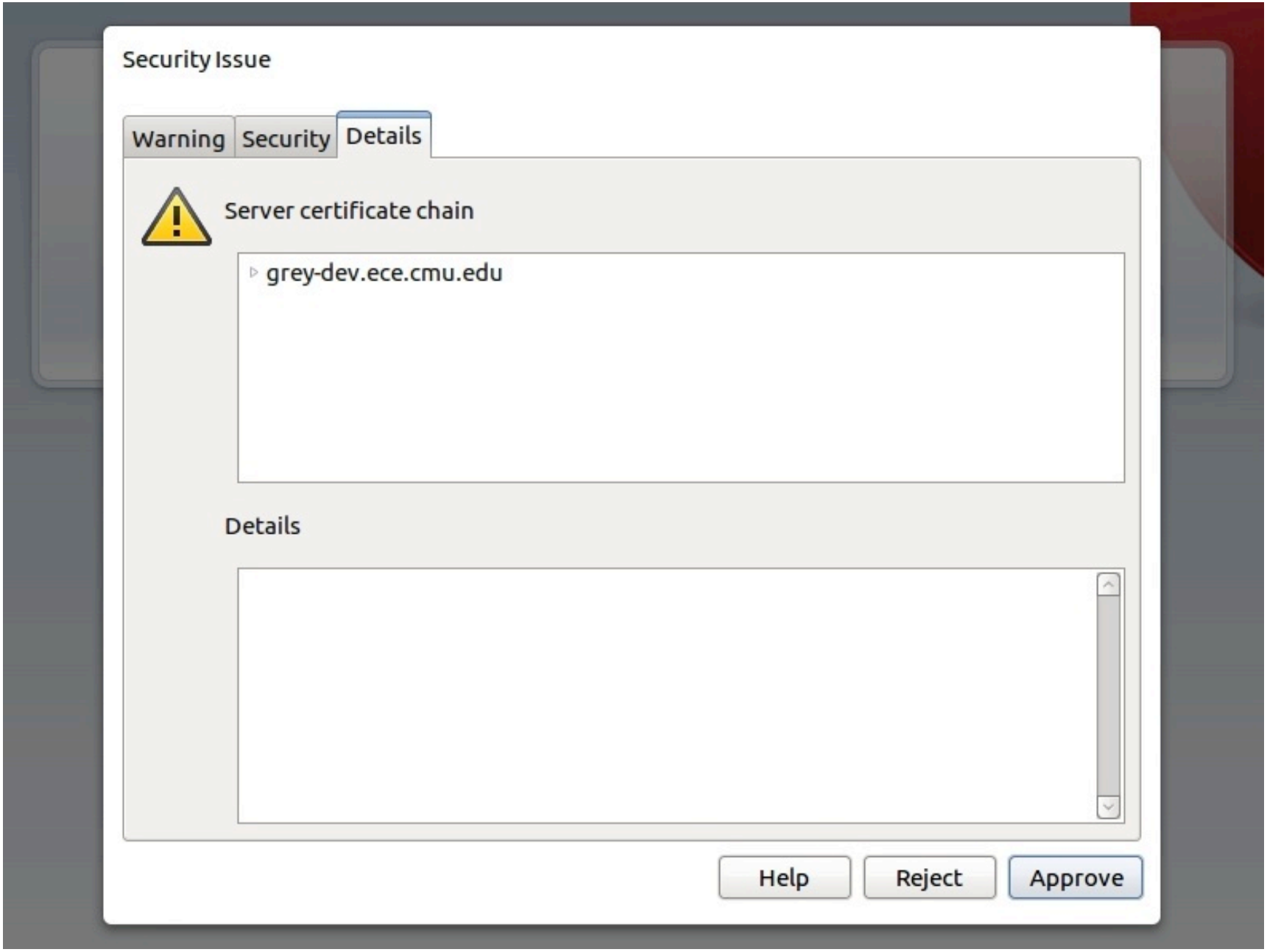
Opera


Security Issue
Warning
Security
Details
The server's certificate chain is incomplete, and the signer(s) are not registered.
Accept?
Server name:
grey-dev.ece.cmu.edu
Help
Reject
Approve

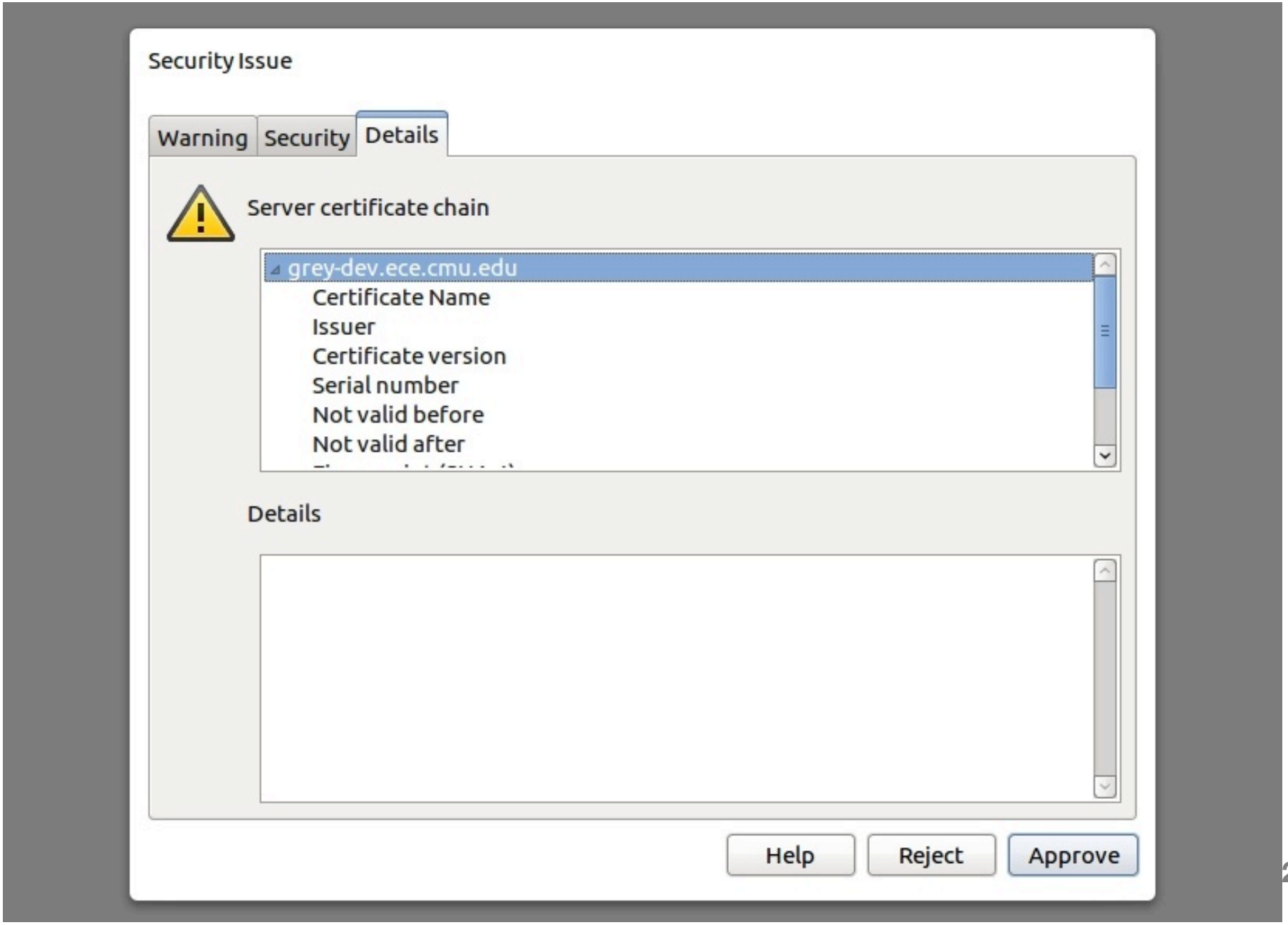
Opera


Security Issue
Warning
Security
Details
Certificate errors:
The certificate for "grey-dev.ece.cmu.edu" is signed by the unknown Certificate Authority "grey-dev.ece.cmu.edu". It is not possible to verify that this is a valid certificate.
Certificate summary
Holder: grey-dev.ece.cmu.edu
Issuer: grey-dev.ece.cmu.edu
Expires: 02/25/2019 02:38:00 PM GMT
Encryption protocol
256 bit AES (DHE_RSA/SHA)
Remember my choice for this certificate
Help
Reject
Approve

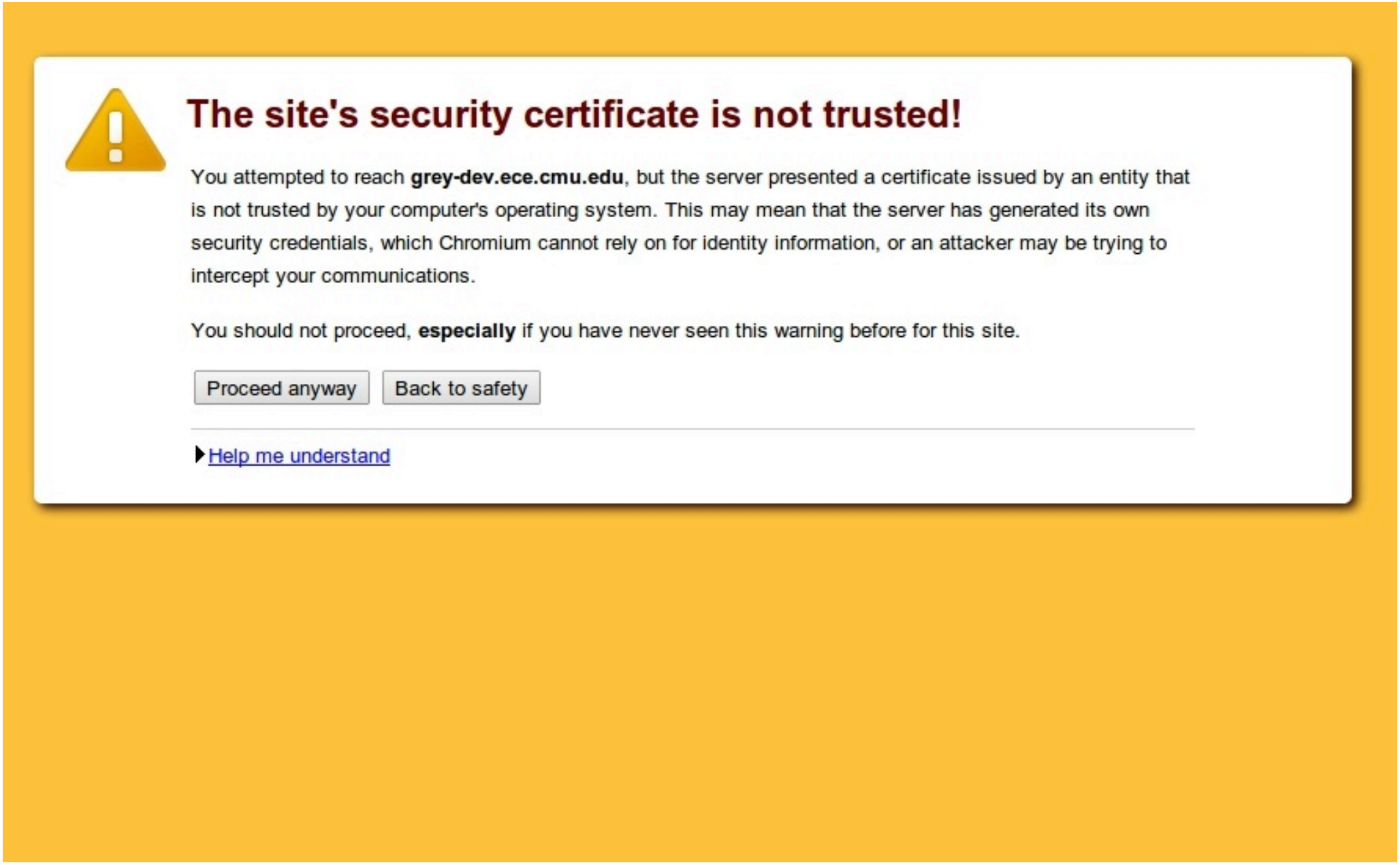
Opera


Security Issue
Warning
Security
Details
Server certificate chain
grey-dev.ece.cmu.edu
Details
Help
Reject
Approve

Opera


Security Issue
Warning
Security
Details
Server certificate chain
grey-dev.ece.cmu.edu
Certificate Name
Issuer
Certificate version
Serial number
Not valid before
Not valid after
Details
Help
Reject
Approve

Chromium


The site's security certificate is not trusted!
You attempted to reach grey-dev.ece.cmu.edu, but the server presented a certificate issued by an entity that is not trusted by your computer's operating system. This may mean that the server has generated its own security credentials, which Chromium cannot rely on for identity information, or an attacker may be trying to intercept your communications.
You should not proceed, especially if you have never seen this warning before for this site.
Proceed anyway
Back to safety
Help me understand

Chromium

You attempted to reach **grey-dev.ece.cmu.edu**, but the server presented a certificate issued by an entity that is not trusted by your computer's operating system. This may mean that the server has generated its own security credentials, which Chromium cannot rely on for identity information, or an attacker may be trying to intercept your communications.

You should not proceed, **especially** if you have never seen this warning before for this site.

Proceed anyway | Back to safety

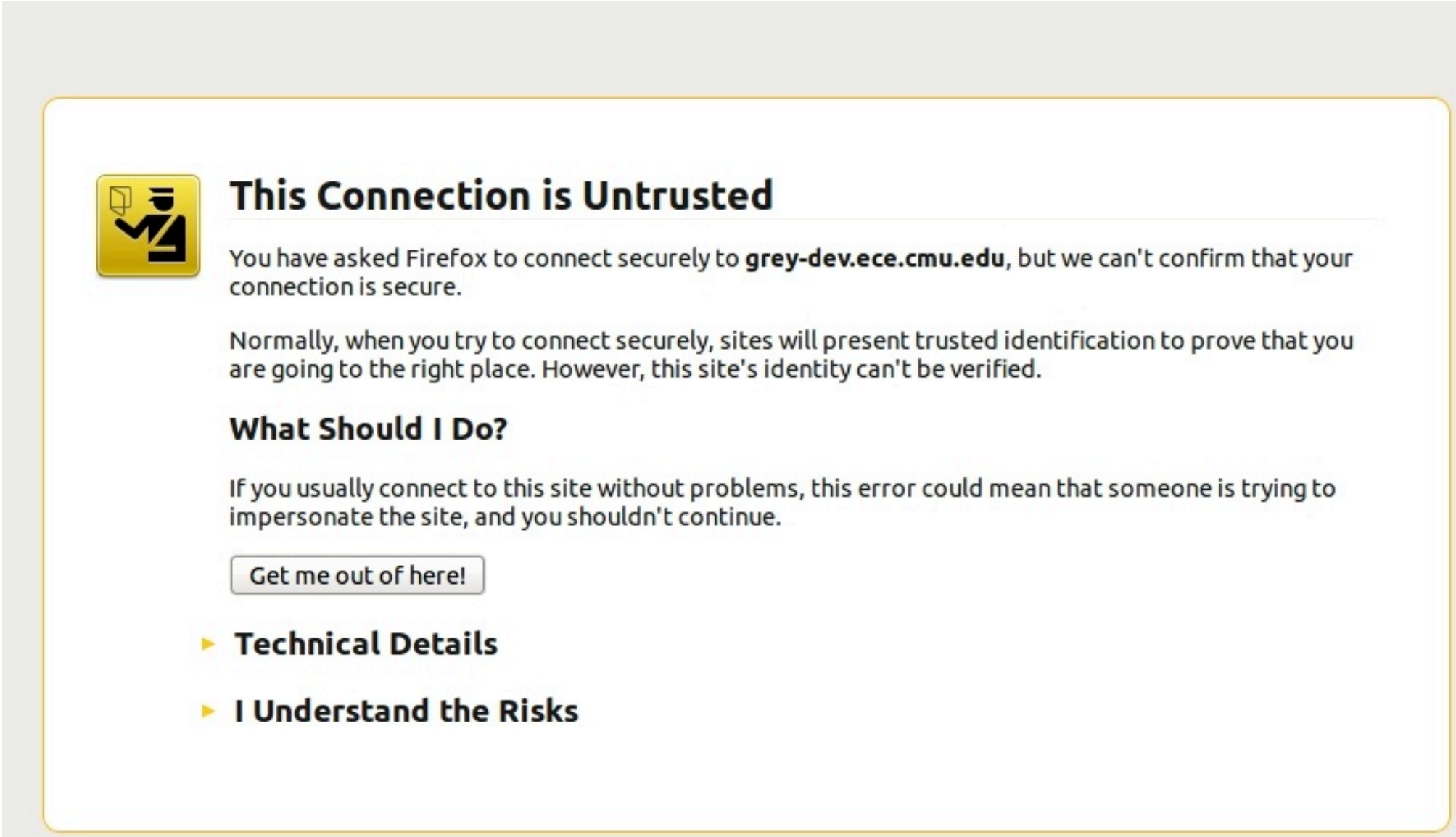
▼ Help me understand

When you connect to a secure website, the server hosting that site presents your browser with something called a "certificate" to verify its identity. This certificate contains identity information, such as the address of the website, which is verified by a third party that your computer trusts. By checking that the address in the certificate matches the address of the website, it is possible to verify that you are securely communicating with the website you intended, and not a third party (such as an attacker on your network).

In this case, the certificate has not been verified by a third party that your computer trusts. Anyone can create a certificate claiming to be whatever website they choose, which is why it must be verified by a trusted third party. Without that verification, the identity information in the certificate is meaningless. It is therefore not possible to verify that you are communicating with **grey-dev.ece.cmu.edu** instead of an attacker who generated his own certificate claiming to be **grey-dev.ece.cmu.edu**. You should not proceed past this point.

If, however, you work in an organization that generates its own certificates, and you are trying to connect to an internal website of that organization using such a certificate, you may be able to solve this problem securely. You can import your organization's root certificate as a "root certificate", and then certificates issued or verified by your organization will be trusted and you will not see this error next time you try to connect to an internal website. Contact your organization's help staff for assistance in adding a new root certificate to your computer.

Mozilla Firefox



Mozilla Firefox

You have asked Firefox to connect securely to **grey-dev.ece.cmu.edu**, but we can't confirm that your connection is secure.

Normally, when you try to connect securely, sites will present trusted identification to prove that you are going to the right place. However, this site's identity can't be verified.

What Should I Do?

If you usually connect to this site without problems, this error could mean that someone is trying to impersonate the site, and you shouldn't continue.

Get me out of here!

Technical Details

grey-dev.ece.cmu.edu uses an invalid security certificate.

The certificate is not trusted because it is self-signed.

(Error code: sec_error_untrusted_issuer)

I Understand the Risks

If you understand what's going on, you can tell Firefox to start trusting this site's identification. **Even if you trust the site, this error could mean that someone is tampering with your connection.**

Don't add an exception unless you know there's a good reason why this site doesn't use trusted identification.

Add Exception...

Activity

- Work in teams of 2-3 to design the user communication related to SSL/TLS for a new browser you're developing: Firepony
  - First, let's come up with our specifications (requirements engineering)