

Two lessons from usable security and its experiments



Vashek Matyas

CRoCS, Masaryk University, Czech Republic

CRoCS

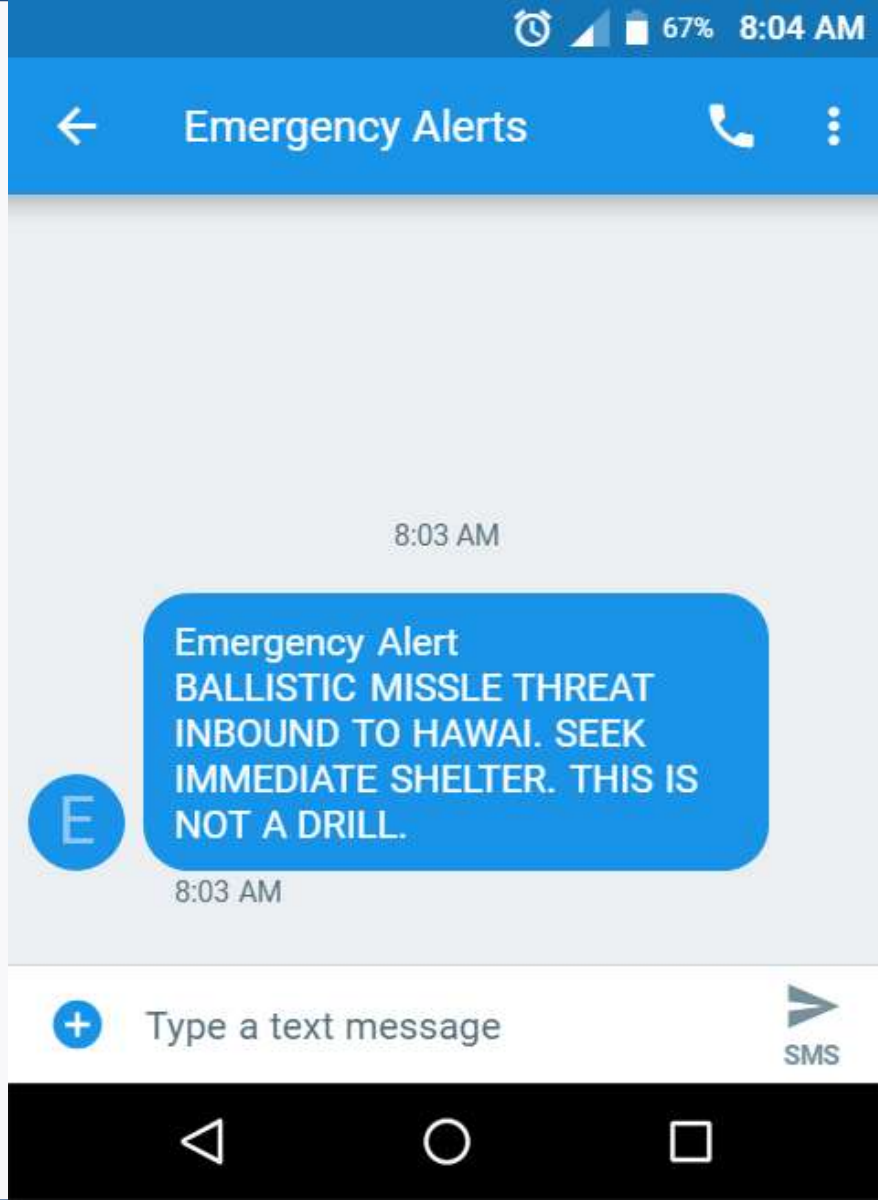
Centre for Research on
Cryptography and Security

*Joint work with Martin Ukrop, Agata Kruzikova, Milan Broz
(first two – Ph.D. in usable security supported by Red Hat
Czech & credits for the slides! 😊)*



13. 1. 2018, Hawaii





8:03

The phone beeps.

A text comes.



38 minutes pass...

EXIT 20 B

EXIT 20 C

Houghtaling St

1/2 MILE

Palama St ↗

MISSILE ALERT
IN ERROR
THERE IS NO THREAT

EXIT 20 C

Palama St

NEXT RIGHT



Cause? Bad warning system UI!

1. State EOC

PACOM (CDW) - STATE ONLY



BMD False Alarm

Amber Alert (CAE) - Kauai County Only

Amber Alert (CAE) Statewide

1. TEST Message

PACOM (CDW) - STATE ONLY

Tsunami Warning (CEM) - STATE ONLY

DRILL-PACOM (DEMO) STATE ONLY

Landslide - Hana Road Closure

Amber Alert DEMO TEST

High Surf Warning North Shores

That was a **usability issue**.

(More precisely, very bad user interface.)

Example of a usable security issue?

Ever heard of encrypted email? (being usable and secure)

Why Johnny Can't Encrypt

A Usability Evaluation of PGP 5.0

ALMA WHITTEN AND J. D. TYGAR

Why Johnny Still Can't Encrypt:

Evaluating the Usability of Email Encryption Software

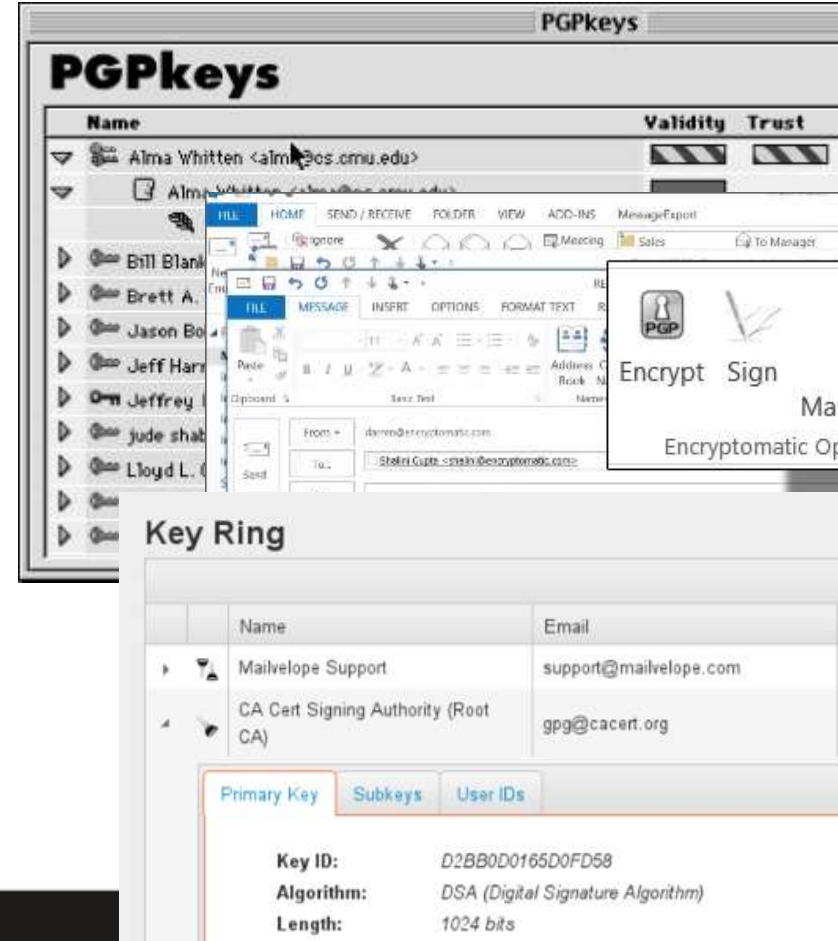
Steve Sheng
Engineering and Public Policy
Carnegie Mellon University
shengx@cmu.edu

Levi Broderick
Electrical and Computer Engineering
Carnegie Mellon University
lpb@ece.cmu.edu

Colleen Alison Koranda
HCI Institute
Carnegie Mellon University
ckoranda@andrew.cmu.edu

Why Johnny Still, Still Can't Encrypt: Evaluating the Usability of a Modern PGP Client

Scott Ruoti, Jeff Andersen, Daniel Zappala, Kent Seamons
Brigham Young University
{ruoti, andersen} @ isrl.byu.edu, {zappala, seamons} @ cs.byu.edu



Ever heard of encrypted email?

(being usable and secure)

15 reasons not to start using PGP

Because of popular demand, here's the collection of reasons to prefer more advanced cryptographic communications tools and stop investing in the old PGP over e-mail architecture, the problem mostly being e-mail rather than PGP.

[Pretty Good Privacy](#) is better than [end](#) it is also better than relying between the mail servers while but is it still a good choice for recommend to people who are a

The text concludes mentioning s this is *not* about not using en intellectual trap of giving backw

1. Downgrade Attack: The



Mayer Mizrachi [Follow](#)

CEO & Founder @Criptext. Magna Cum Hack — Picota 2016.

May 18 · 7 min read

It's Time To Drop PGP

"Email is no longer a secure communication medium"
Schinzel

Schneier on Security

[Blog](#)

[Newsletter](#)

[Books](#)

[Essays](#)

[News](#)

[Talks](#)

[Academic](#)

[About Me](#)

[Blog](#) >

Giving Up on PGP

Filippo Valsorda wrote an [excellent essay](#) on why he's giving up on PGP. I have long believed PGP to be more trouble than it is worth. It's hard to use correctly, and easy to get wrong. More generally, e-mail is inherently difficult to secure because of all the different things we ask of it and use it for.

Valsorda has a different complaint, that its long-term secrets are an unnecessary source of risk:

[About Bruce S](#)



Usable security for...

End-users

IT professionals

Usable security for...

End-users

- How do we nudge users to choose secure passwords?
- Which biometric is the most usable? (w.r.t. Its security)

IT professionals

Usable security for...

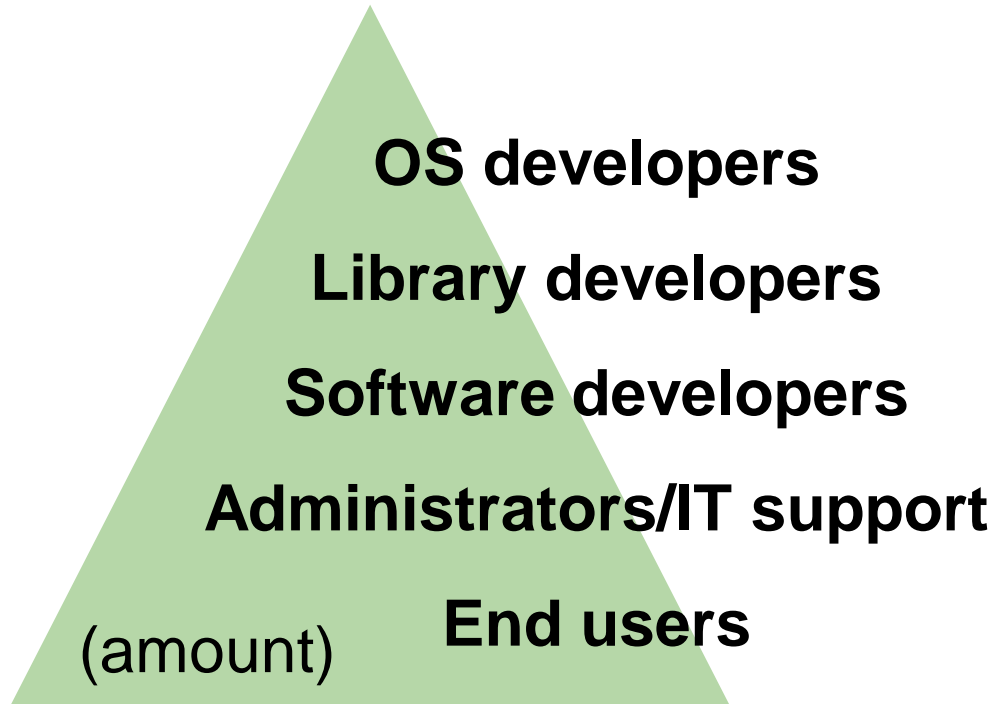
End-users

- How do we nudge users to choose secure passwords?
- Which biometric is the most usable? (w.r.t. its security)

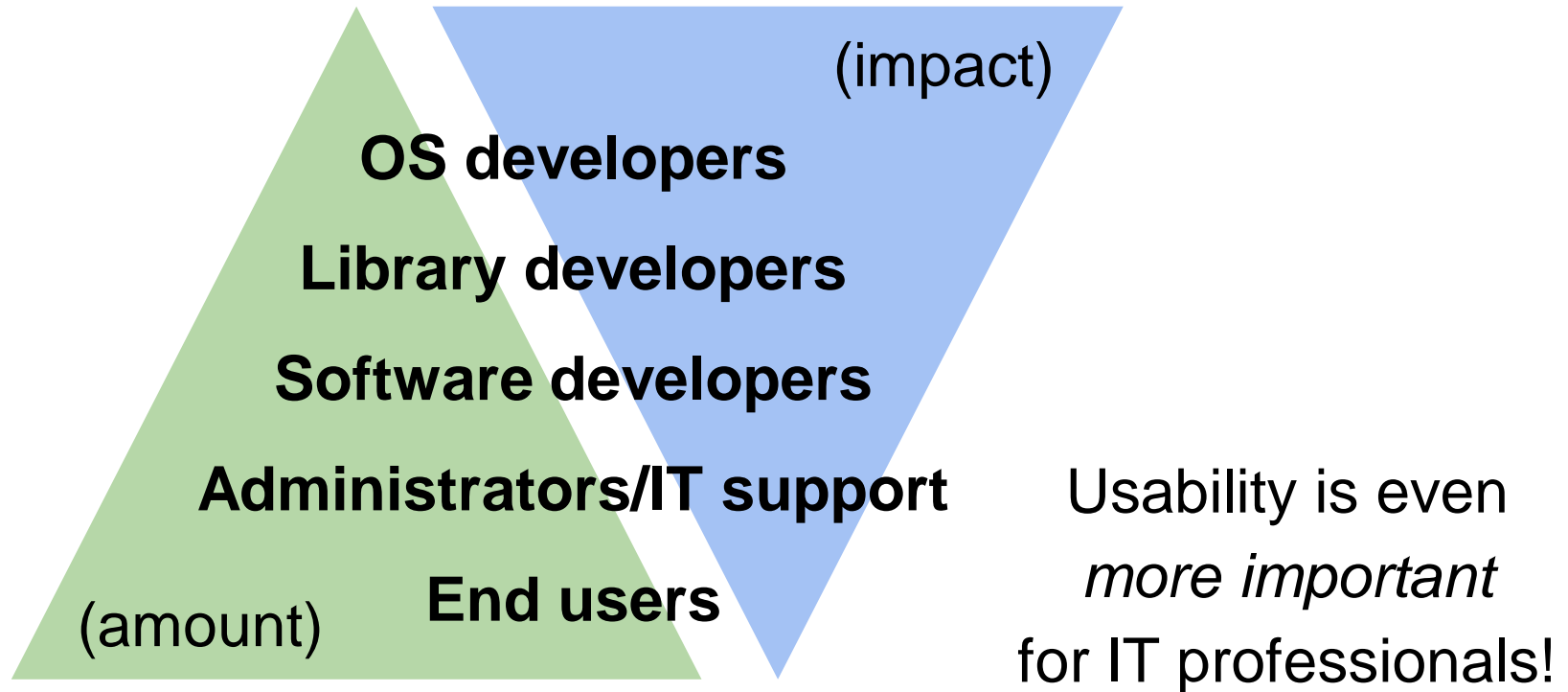
IT professionals

- Is the intuitive configuration of the server also secure?
- Do developers understand security error messages?

The impact pyramid



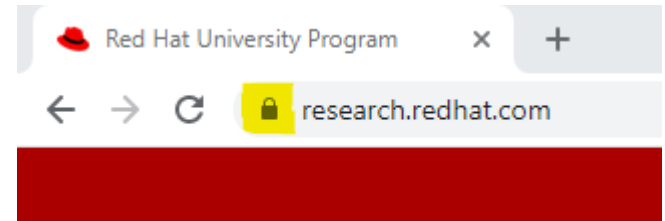
The impact pyramid



Our focus: Usable work with certificates

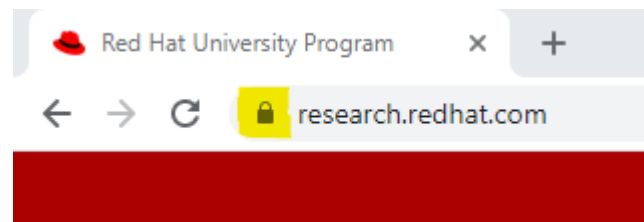
Our focus: Usable work with certificates

- Certificates are ubiquitous (think about TLS)



Our focus: Usable work with certificates

- Certificates are ubiquitous (think about TLS)



- TLS API is notoriously complicated

The Most Dangerous Code in the World: Validating SSL Certificates in Non-Browser Software

Martin Georgiev
The University of Texas
at Austin

Rishita Anubhai
Stanford University

Subodh Iyengar
Stanford University

Dan Boneh
Stanford University

Suman Jana
The University of Texas
at Austin

Vitaly Shmatikov
The University of Texas
at Austin

DEVCONF.cz 2017 experiment

- Research booth
- 87 participants interacting with OpenSSL
- Usability of certificate generation and validation



DEVCONF.cz 2017 experiment

- Selected results
 - Perceived success (87%) vs. reality (45%)
 - Default arguments matter a lot!
 - Unintuitive manual page names

DEVCONF.cz 2017 experiment

- Selected results
 - Perceived success (87%) vs. reality (45%)
 - Default arguments matter a lot!
 - Unintuitive manual page names
- Two patches in upstream OpenSSL
- Academic publication at RSA-CT 2018

RSA® Conference | Where the world
talks security

Going further: Error understanding

- We now know OpenSSL usability is bad.
- Do people understand the errors?
 - Is the certificate still trustworthy?
 - Is the error severe?

→ Let's make a new experiment!

DEVCONF.cz 2018 experiment

- Research booth
- 75 participants seeing five certificate errors
- Understanding / trust
- Comparing existing and our “improved” docs





The task at DevConf 2018

- Connect to an authentication server
(GitHub, Fedora Project, Google, Microsoft, Facebook)
 - 1) Try to understand what is wrong with the cert.
 - 2) Decide how much you trust the certificate.

(expired, OK, name constrained, hostname mismatch, self-signed)

DEVCONF.cz 2018 experiment

- Selected results
 - Trust is far from binary (“I kind of trust that ...”)
 - Some cases are over-trusted or poorly understood
 - Redesigned documentation works the same/better

DEVCONF.cz 2018 experiment

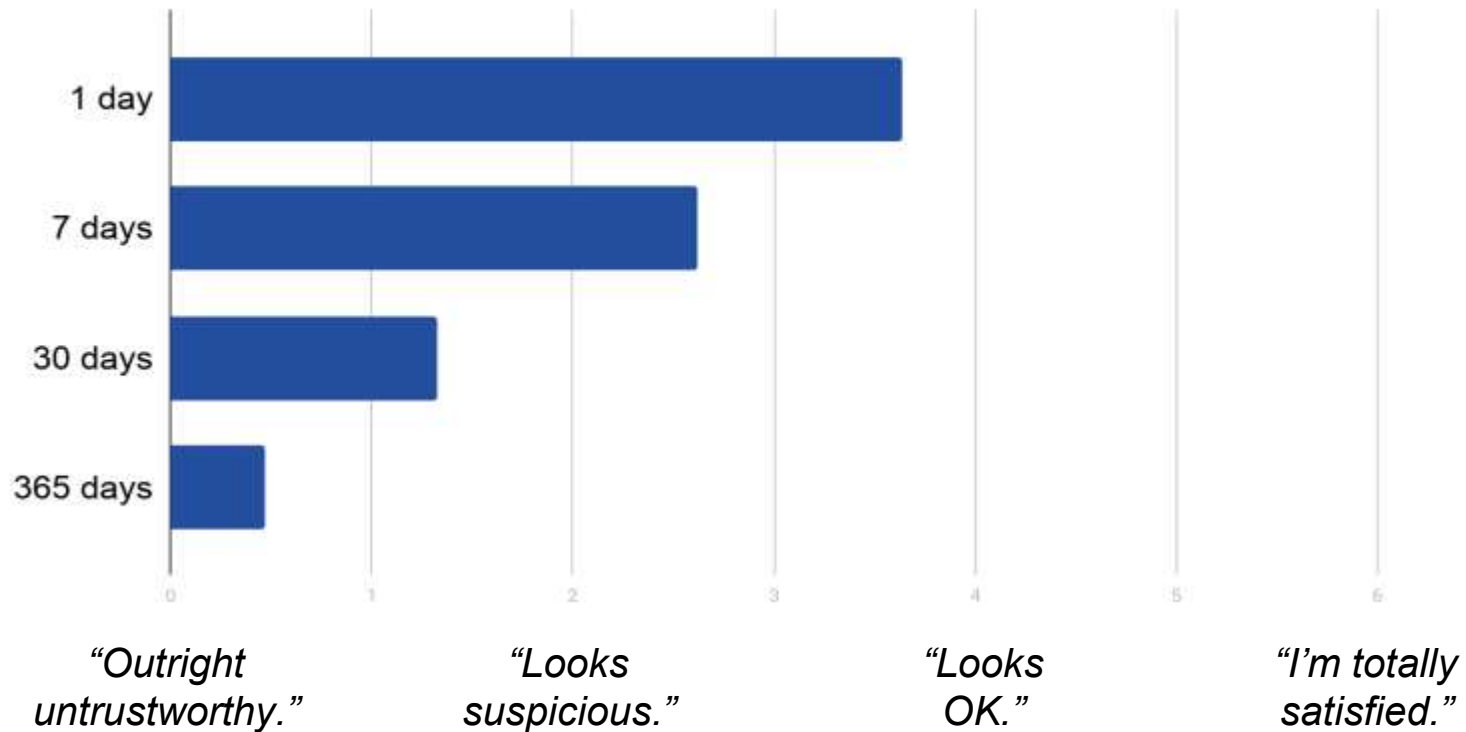
- Selected results
 - Trust is far from binary (“I kind of trust that ...”)
 - Some cases are over-trusted or poorly understood
 - Redesigned documentation works the same/better
- A small patch in upstream OpenSSL
- Academic publication at ACSAC 2019



ACSAC 2019

December 9-13, 2019 • San Juan

Trust in expired certificates



Going further: x509errors.org






- We now know what does not work.
- Let's fix it! (Or at least a bit of it.)
 - Consolidate and map existing errors from multiple libraries
 - Create better documentation

x509errors.org

Usable X.509 errors: OpenSSL

Our goal is to simplify the ecosystem by consolidating the errors and their documentation (similarly to [web documentation](#)) and better explaining what the validation errors mean.

Correctly validating X.509 certificates turns out to be pretty complicated (e.g., [Georgiev2012](#), [Ukrop2019](#)). Yet certificate validation is crucial for secure communication on the Internet (think [TLS](#)).

For every error, we aim to provide our redesigned documentation (), an example certificate (), original documentation provided by the library ( , unused or deprecated errors denoted by ). Furthermore, we provide links to corresponding errors from other libraries (). In the future, we plan on adding error frequencies based on IP-wide scans and elaborating on the consequences of individual errors.

[See more in FAQ](#)

 **x509errors.or**




Basic extension errors

Errors related to extensions in general or to the BasicConstraints standard extension.

Relevant links: [Certificate Extensions](#) (RFC 5280), [BasicConstraints Extension](#) (RFC 5280)

> X509_V_ERR_UNSUPPORTED_EXTENSION_FEATURE 

> X509_V_ERR_INVALID_CA  

> X509_V_ERR_PATH_LENGTH_EXCEEDED   

> X509_V_ERR_UNHANDLED_CRITICAL_EXTENSION   

> X509_V_ERR_UNHANDLED_CRITICAL_CRL_EXTENSION

> X509_V_ERR_INVALID_EXTENSION  

Name related errors

Errors signaling problems with either hostname verification, NameConstraints standard extension or IP Address D

Relevant links: [NameConstraints extension](#) (RFC 5280), [IP Address Delegation extension](#) (RFC 3779), [Certificate C](#)

 **x509errors.or**

▼ X509_V_ERR_INVALID_CA



Original documentation:

A CA certificate is invalid. Either it is not a CA or its extensions are not consistent with the supplied purpose. ([source](#))

Original error message:

invalid CA certificate ([source](#))

Example certificates

Below you can download one or more example malformed certificates causing X509_V_ERR_INVALID_CA in OpenSSL. If you are interested in generating these certificates yourself, see the corresponding generating script for each case on the project Github.

- Case [issuer-ca-false](#) (see the [generation script](#))

Corresponding errors

What validation errors do other libraries give for certificates causing X509_V_ERR_INVALID_CA in OpenSSL? Below, you can see the basic overview based on the example certificates from the previous section. (*The list may be incomplete.*)

- GnuTLS: [GNUTLS_CERT_SIGNER_NOT_CA](#)
- Botan: [CA_CERT_NOT_FOR_CERT_ISSUER](#)
- Mbed TLS: [MBEDTLS_X509_BADCERT_NOT_TRUSTED](#)
- OpenJDK: [PKIX_PATH_VALIDATION_FAILED](#), [NOT_A_CA_CERTIFICATE](#)

➤ X509_V_ERR_PATH_LENGTH_EXCEEDED

x509errors.or

▼ X509_V_ERR_PATH_LENGTH_EXCEEDED



Redesigned documentation:

The allowed length of the certification path was exceeded.

Explanation

Certification Authorities (CAs) can mandate the maximal length of the trusted certificate chains below their certificate. This is done using the `pathLenConstraint` field in the `basicConstraints` extension. If the certificate chain created during validation is longer than this limit, the validation fails due to the violated path length constraint. This limit includes only intermediate certificates – the first (CA) and the last (endpoint) certificates are excluded.

Security perspective

An exceeded certificate path length signifies that one of the sub-authorities issued a certificate it was not allowed. Therefore, the CA or one of the sub-authorities may not be trustworthy.

Next steps

Inspect the certificate chain to find the `pathLenConstraint` in the `basicConstraints` extension that was violated. Inform the (sub-)authority issuing this certificate about the violation lower in the certificate chain.

Original documentation:

The basicConstraints pathlength parameter has been exceeded. [\(source ↗\)](#)

Original error message:

path length constraint exceeded [\(source ↗\)](#)

Example certificates

Below you can download one or more example malformed certificates causing X509_V_ERR_PATH_LENGTH_EXCEEDED.

x509errors.or

DEVCONF.cz 2020 experiment

- Developer survey
- 180 people evaluating two docs versions
- Length? Content? Ambiguity? Bloat?
- Understanding? Satisfaction?



Original documentation (OpenSSL)

X509_V_ERR_UNABLE_TO_GET_CRL_ISSUER

Unable to get CRL issuer certificate.

X509_V_ERR_UNHANDLED_CRITICAL_EXTENSION

Unhandled critical extension.

X509_V_ERR_KEYUSAGE_NO_CRL_SIGN

Key usage does not include CRL signing.

Redesigned documentation

X509_ERR_UNHANDLED_CRITICAL_EXTENSION

Either critical extension was not recognized, or information in critical extension could not be processed.

Explanation

Certificate extensions can be used for incorporating additional information into a certificate. The extensions can be critical or non-critical. All extensions marked as critical must be processed. If a system, which processes a certificate, cannot recognize a critical extension, it must reject the certificate. It has to reject the certificate also when it recognizes the critical extension, but it cannot process the information contained in the extension.

Security perspective

An extension can carry arbitrary information, and marking it as critical means that it is crucial to process it. If it cannot be processed, there is a security risk that a certificate's key will be used in a manner it must not be, e.g., that a certificate's key will be used for another purpose that it was aimed or that a Certification Authority will issue a certificate for subject name for which it is not allowed to issue certificates, or many other security risks.

What to do

If you are responsible for the certificate, make sure that only necessary extensions are marked as critical and that the values of critical extensions are meaningful. If you are not responsible for the certificate, you can check the critical extensions and the values which contain, but it is not recommended to continue processing the certificate.

Consequences

If you ignore critical extensions that cannot be processed, it may result in unauthorized use of the certificate.

DEVCONF.cz 2020 experiment

- New documentation
 - Decreased incompleteness, ambiguity, inconsistency
 - Slightly increased bloat, tangle
 - Increased understanding, satisfaction, helpfulness

DEVCONF.cz 2020 experiment

- New documentation
 - Decreased incompleteness, ambiguity, inconsistency
 - Slightly increased bloat, tangle
 - Increased understanding, satisfaction, helpfulness
- Overall opinions
 - Wanted slightly shorter than our (structured!)
 - 89% participants preferred the redesign

DEVCONF.cz 2020 experiment

- New documentation
 - Decreased incompleteness, ambiguity, inconsistency
 - Slightly increased bloat, tangle
 - Increased understanding, satisfaction, helpfulness
- Overall opinions
 - Wanted slightly shorter than our (structured!)
 - 89% participants preferred the redesign
- We validated new designs and now create more!

Next step: Get to official documentation



[Home](#) [Blog](#) [Downloads](#) [Docs](#) [News](#) [Policies](#) [Community](#) [Support](#)

X509_STORE_CTX_get_error

NAME

X509_STORE_CTX_get_error, X509_STORE_CTX_set_error, X509_STORE_CTX_get_error_depth, X509_STORE_CTX_set_error_depth, X509_STORE_CTX_get_current_cert, X509_STORE_CTX_set_current_cert, X509_STORE_CTX_get0_cert, X509_STORE_CTX_get1_chain, X509_verify_cert_error_string - get or set certificate verification status information

master manpages

[Commands](#)

[Libraries](#)

[File Formats](#)

[Overviews](#)

This manpage

3.0 version

Authentication of IT Professionals in The Wild – A Survey

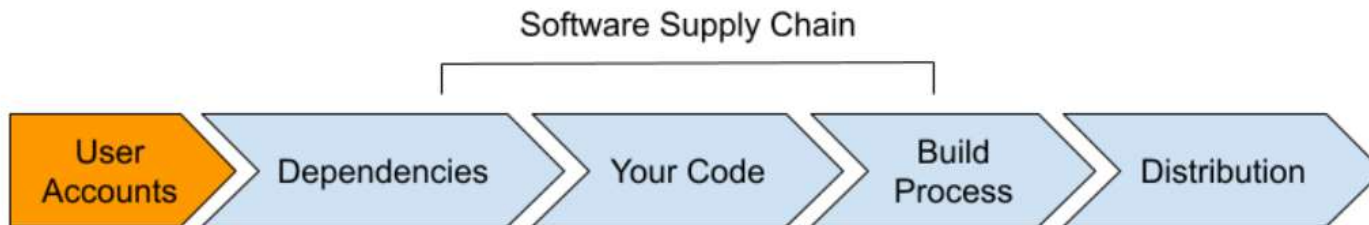


Joint work with Agata Kruzikova & Milan Broz



Why user authentication in GitHub?

- Open source → source for commercial companies
- Supply chain attack → importance of trust
- Independent developers → no IT security policy



Source: GitHub blog [Raising the bar for software security: next steps for GitHub.com 2FA](#)

User authentication options in GitHub

- 1st factor of authentication
 - Login and password
- 2nd factor of authentication
 - Authentication app (SW token)
 - Security keys (HW token)
 - SMS number (SMS code)
- Recovery options
 - Recovery codes
 - Fallback SMS number
 - Recovery tokens (Facebook)



Source: docs.github.com

Study procedure

- Quantitative questionnaire sent via mailing lists to Red Hat employees
 - Demography
 - Usage and perception of a GitHub account
 - Experience, usability and security perception of authentication
 - Task: authentication log
- Data collected in November 2020

Main findings

- 2FA mostly used (by 81% of participants)
 - Mostly SW and HW tokens
 - Mostly recovery codes for fallback authentication
 - Methods mostly perceived as (rather) usable and secure
- Facebook as fallback method – not evaluated
 - 57% not evaluated
 - 70% perceived as (rather) insecure

Two-factor and fallback authentication

Experience	SW token	HW token	SMS code
Current	69% (N=57)	31% (26)	18% (15)
Past	2% (2)	2% (2)	14% (12)
None	24% (20)	58% (48)	57% (47)
Not remember	2% (1)	1% (2)	4% (3)
Missing answers	2% (2)	7% (6)	7% (6)

Experience	Recovery codes	SMS code	Login via Facebook
Already used	19% (N=16)	15% (12)	1% (1)
Activated	49% (41)	29% (24)	6% (5)
Not activated	4% (3)	27% (22)	54% (45)
Not remember	5% (4)	6% (5)	4% (3)
Missing answers	23% (19)	24% (20)	35% (29)

Limitations

- Sample
 - Small sample – hardly achievable
 - 252 clicks on the survey link registered
 - 83 participants (10 000 addressed people)
 - Mostly software engineers (75%)
 - Office: 33% US, 29% CZ, 33% other (5% no answer)
 - Self-selection bias
- Self-reported data

2FA enforcement

- Only 16.5% of (active) GitHub users use 2FA
- GitHub – 2FA enforcement for contributors by the end of 2023
- See more at [Software security starts with the developer: Securing developer accounts with 2FA | The GitHub Blog](#)

2FA enforcement

- Only 16.5% of (active) GitHub users use 2FA
- GitHub – 2FA enforcement for contributors by the end of 2023
- See more at [Software security starts with the developer: Securing developer accounts with 2FA | The GitHub Blog](#)
- Why so few users of 2FA?

Where to go next

- How users perceive the 2FA usage enforcement?
 - Company/maintainer enforcement
 - GitHub enforcement

Where to go next

- How users perceive the 2FA usage enforcement?
- Why users have not started to use 2FA yet?
 - Do users consider 2FA as important?
 - Do users perceived other security measures as sufficient?

Where to go next

- How users perceive the 2FA usage enforcement?
- Why users have not started to use 2FA yet?
- Is perception different for users with different rights/responsibilities?

Where to go next

- How users perceive the 2FA usage enforcement?
- Why users have not started to use 2FA yet?
- Is perception different for users with different rights/responsibilities?
- Is the list of 2FA methods sufficient?

Takeaways

- Most of our participants already used 2FA in 2020
 - Yet only 16.5% GitHub users actually used 2FA in 2022 according to GitHub
- Facebook
 - Least secure method
 - Most missing values

Wrap-up of the session

- Usable security research...

Testing the usability of security tools with developers, admins, etc.

- ...focusing on certificate validation...

Certificates are widely used for ensuring security (e.g. TLS).

Error messages and documentation are poor.

- ...aiming for real-world impact.

Not just proof of concept – designing better documentation, validating with IT professionals, trying to get upstream to OpenSSL.

Now go and make your software usable!

(And secure!)



Got interested? Let me know :-).

CRoCS

Centre for Research on
Cryptography and Security

Vashek Matyas, matyas@fi.muni.cz

CRoCS, Masaryk University, Czech Republic

