

# Analyzing Compliance and Complications of Integrating Internationalized X.509 Certificates

*Mingming Zhang<sup>1</sup> and Jinfeng Guo<sup>2</sup>, Yiming Zhang<sup>3</sup>, Shenglin Zhang<sup>2</sup>, Baojun Liu<sup>3</sup>, Hanqing Zhao<sup>3</sup>,  
Xiang Li<sup>2</sup>, Haixin Duan<sup>3,4</sup>*

*<sup>1</sup>Zhongguancun Laboratory <sup>2</sup>Nankai University <sup>3</sup>Tsinghua University <sup>4</sup>Quancheng Laboratory*

Presenter: Mingming Zhang  
Zhongguancun Laboratory

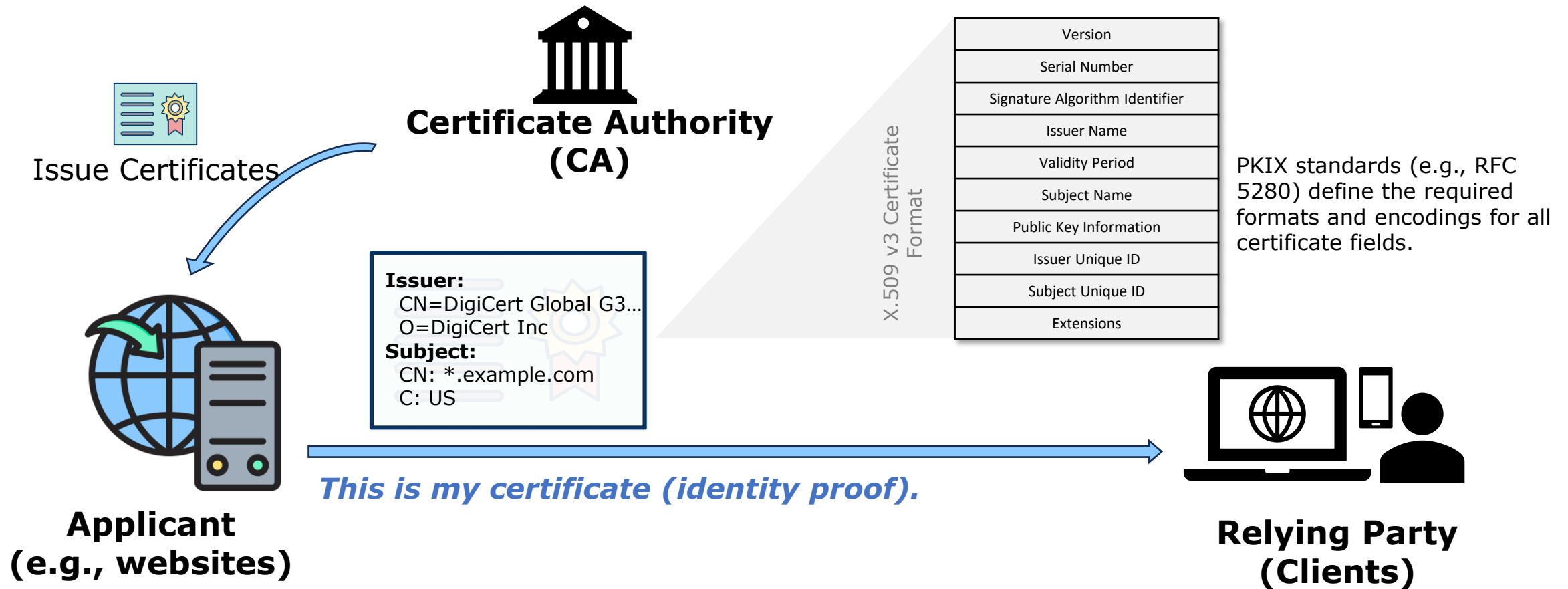
2026.03

# PKI and X.509 Certificates

➤ **Public Key Infrastructure (PKI)** is the foundation of trust.

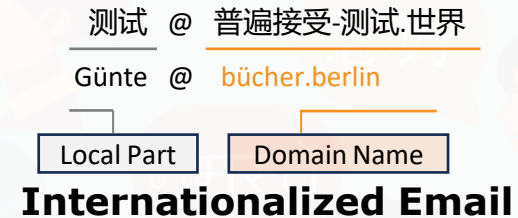
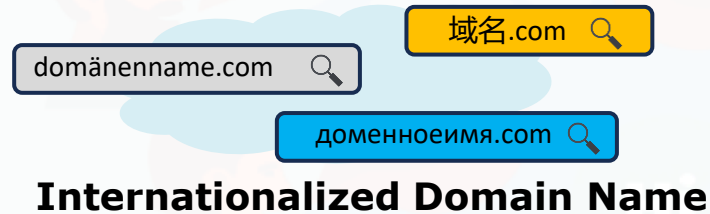
➤ **X.509 Certificates:**

*Binding identities (e.g., hostnames) to cryptographic keys.*



# Unicerts: X.509 Certificates with Internationalized Contents

- **Universal Acceptance (UA)** : An initiative for a truly multilingual and digitally inclusive Internet, promoting Internet applications/systems (e.g., PKI) adopt characters beyond ASCII.



- **(Uni)certs**: X.509 certificates containing internationalized contents, e.g., domain names (IDNs), resource identifiers (IRIs), or multilingual text with non-printable-ASCII set.

**xn--eqrt2g.com**

Subject Name  
Common Name xn--eqrt2g.com

Issuer Name  
Country or Region US  
Organization Let's Encrypt  
Common Name E6

Serial Number 05 B6 F7 6A 5E 07 C6 51 2B 91 7A 9C 9F F8 7D 66 AB 16  
Version 3  
Signature Algorithm ECDSA Signature with SHA-384 ( 1.2.840.10045.4.3.3 )  
Parameters None

**shop.musik-gillhaus.de**

Subject Name  
Country or Region DE  
Locality Freiburg  
Organization Musik Gillhaus Gesellschaft mit beschränkter Haftung  
Common Name shop.musik-gillhaus.de

Issuer Name  
Country or Region US  
Organization DigiCert Inc  
Organizational Unit www.digicert.com  
Common Name DigiCert High Assurance CA-3

**flystaff.tap.pt**

Subject Name  
Country or Region PT  
Locality Lisboa  
Organization MEGASIS, SOCIEDADE DE SERVIÇOS E ENGENHARIA INFORMÁTICA  
S.A.  
Organizational Unit Megasis

Issuer Name  
Country or Region US  
Organization DigiCert Inc  
Organizational Unit www.digicert.com  
Common Name DigiCert High Assurance CA-3

Punycode of the Internationalized Domain Name (IDN): 域名.com

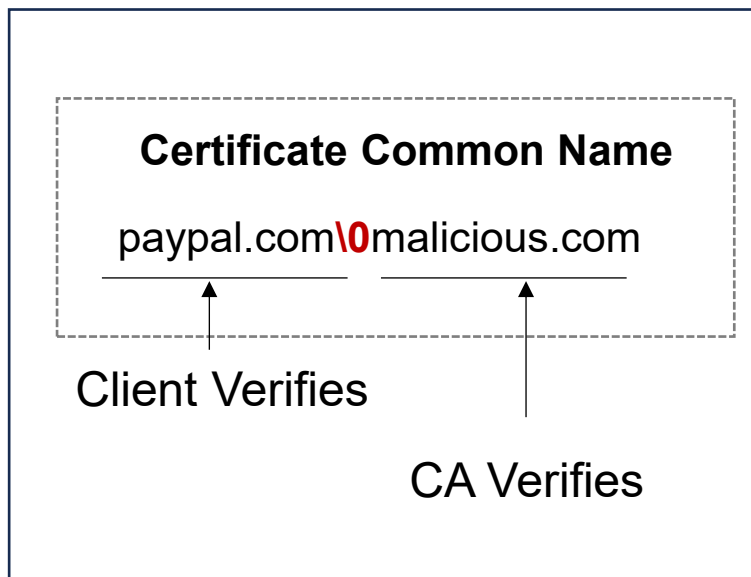
Contain non-printable-ascii characters

# Motivation: Why Unicerts Break Things

- Incorporating Unicode **complicates** the issuance, parsing, and validation lifecycle of X.509 certificates, leading to security or usability issues.

- **Real-World Precedents:**

Improper use of Unicode in certificates has caused:



**Certificate spoofing**

**MyF5** SUPPORT MY PRODUCTS & PLANS

Known Issue

**Archived - K81239824: The X509 iRules commands may incorrectly parse SSL certificate attributes**

Published Date: Apr 19, 2019 Updated Date: Feb 22, 2023

Applies to:

BIG-IP LTM  
12.1.4, 12.1.3, 12.1.2, 12.1.1, 12.1.0, 12.0.0, 11.5

**Impact**

The system may fail to pass properly formatted certificate information to the servers.

**Symptoms**

As a result of this issue, you may encounter one or more of the following symptoms:

- The BIG-IP system fails to pass properly formatted certificate information to the server.
- You observe messages similar to the following example in the `/var/log/itm` file, displaying incorrectly parsed attributes:  
tm1[12345]: Rule /Common/test <HTTP\_REQUEST>: BEFORE subject: serialNumber=1234567892020,CN=John Smith,SN=Smith,GN=John,OU=Individuals,ST=Washington,C=US
- tm1[12345]: Rule /Common/test <HTTP\_REQUEST>: BEFORE issuer: CN=TRUST Root,2.5.4.97=#123456415

**Incorrect attribute parsing**

Home > Notes > VU#794340

OpenSSL 3.0.0 to 3.0.6 decodes some punycode email addresses in X.509 certificates improperly

**Vulnerability Note VU#794340**

Original Release Date: 2022-11-01 | Last Revised: 2024-03-08

FortiGuard Labs Research Services Threat Intelligence Support Resources

PSIRT

**OpenSSL3 CVE-2022-3602 CVE-2022-3786 vulnerabilities**

**Buffer overflows**

**The gap:** While X.509 supports Unicode, the UA readiness of PKI's core mechanisms (issuance, parsing) has been **largely unexamined**.

# Our Research Questions

- We conducted the first large-scale measurement and empirical study of Unicerts



## **RQ1 Issuance Compliance**

*Have CAs issued Unicerts in compliance with complex standard requirements?*



## **RQ2 Parsing Accuracy**

*Do mainstream TLS implementations parse Unicerts according to normative constraints?*



## **RQ3 Real-World Impact**

*What are the security and usability impacts of noncompliant issuance and parsing flaws?*

# RQ1: Issuance Compliance of Unicerts

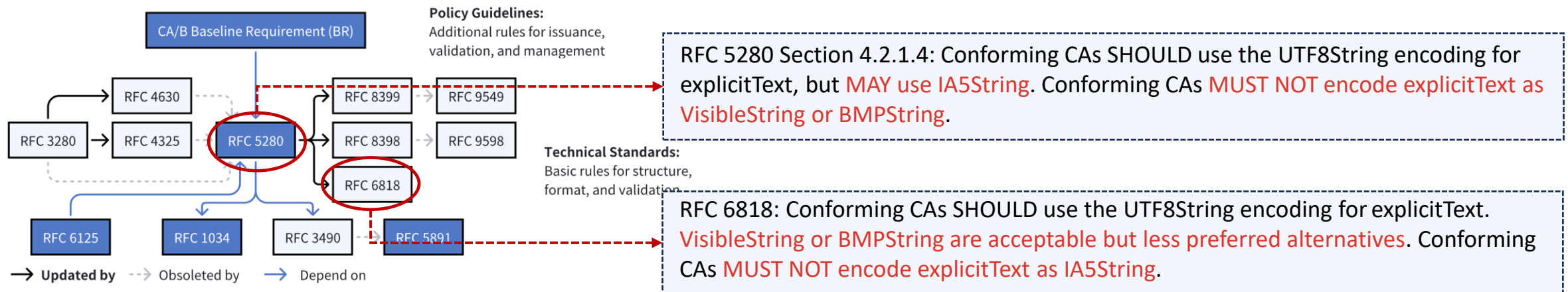


**How do we establish definitive and comprehensive normative requirements for Unicert compliance?**

## ➤ Mastering the specifications

### □ Challenges

1. Standards (e.g., X.509 & updates, DNS & IDN specs, CA/B BRs) are interdependent and evolving.
2. X.509 certificate fields involve complex formats, structures, and encoding rules expressed in diverse representations (e.g., natural language, ASN.1, tables).



**E.g., Requirements evolve through periodic updates and revisions.**

# RQ1: Issuance Compliance of Unicerts

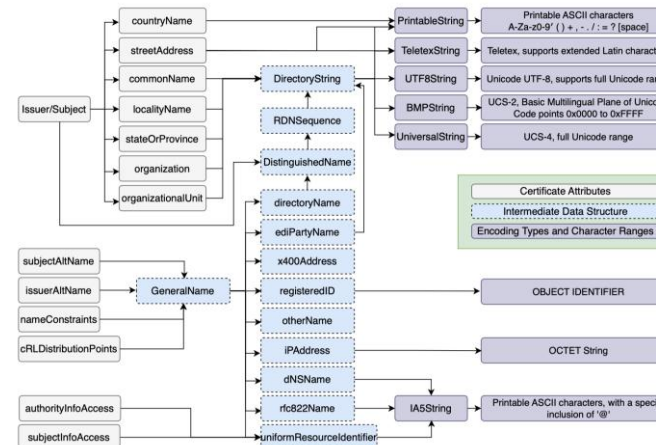
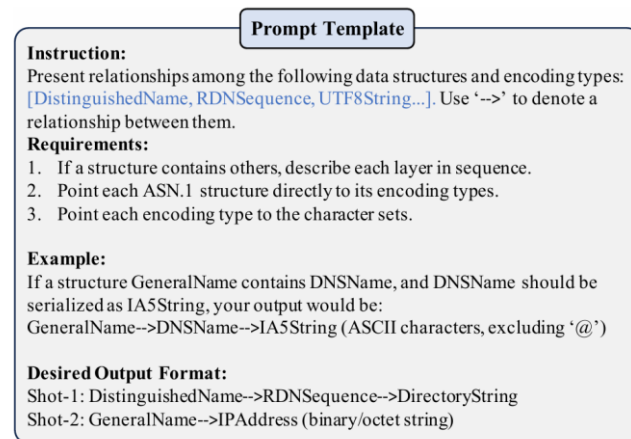
## ➤ Mastering the specifications.

### ❑ Our solution:

- Employing LLM to navigate complex requirements and constraints.

*PKIX specs (RFC 5280), updates (RFC 9549, 9598, 6818), references (RFC 3490, 1034, 3454), dependent standards (RFC 6125, IDNA suites), and CA/B BRs.*

- **Prompt example:** identifying encoding, structure, and character constraints for cert fields.

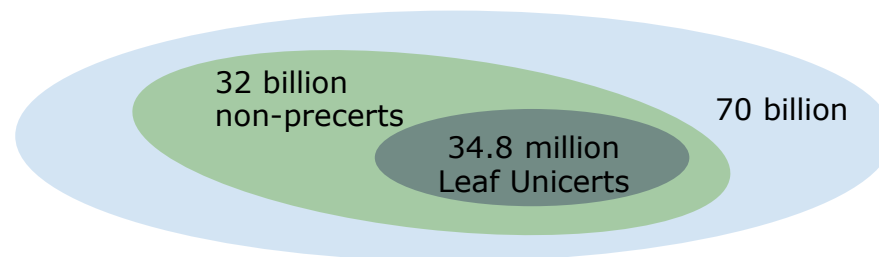


**Results:** Extracted 95 rules for 36 Unicode-related fields, with 50 missing from existing compliance checks.

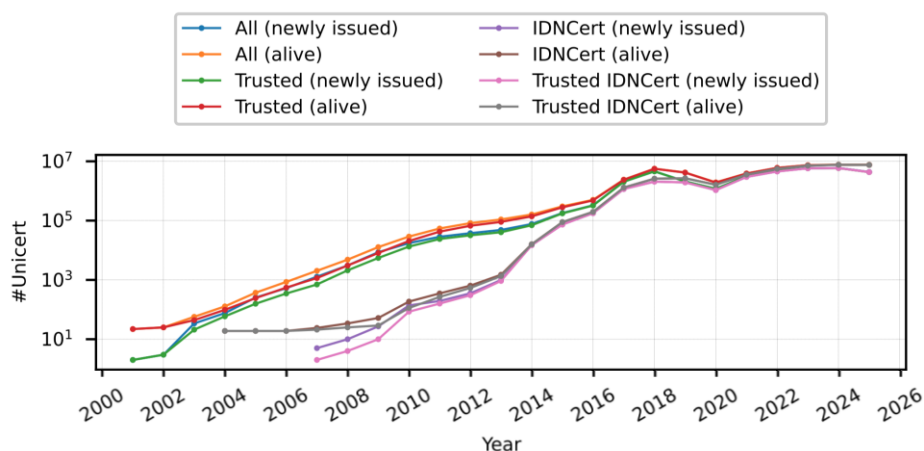
# Overview of Unicert Issuance

## ➤ Measurement

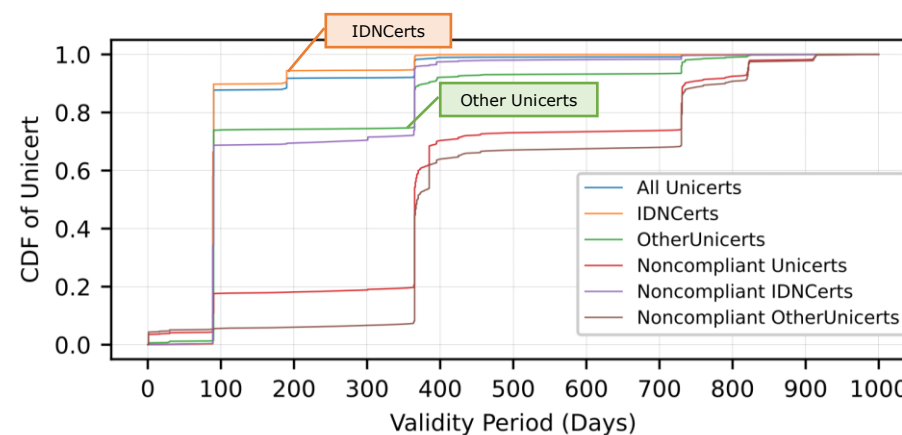
- ❑ Checking **34.8 Million** Unicerts from a 70 Billion CT certificate dataset, against the 95 rules.



- ❑ **Scale of Adoption:** The Unicerts were issued by **4,528** CA certificates across **698** issuer organizations (97.6% from the trusted CAs).



Unicert issuance shows an upward trend.



Unicerts containing only IDNs have shorter validity periods.

# Quantifying Real-world Noncompliance

## ➤ We identified **249K noncompliant Unicerts** from the **CT dataset**.

- ❑ 65.3% from publicly trusted CAs; 21.1% from issuers with limited trust; 13.6% from untrusted issuers

## ➤ **Common issue types**

### ❑ **T1. Improper character checks (43K Unicerts):**

- Malformed strings: e.g., non-printable characters in PrintableString
- Disallowed characters: e.g., control characters in UTF8String

### ❑ **T2. Lack of value normalization (3 certs):**

- ExplicitText (UTF8String) not normalized to Unicode Normalization Form C (NFC)
- IDNs not normalized to NFC after converting Punycode (xn-- string) to Unicode

### ❑ **T3. Invalid format/structure (206K certs ):**

- Illegal format: basic formatting errors, which can hinder parsing
- Invalid encoding: e.g., encoding CN with TeletexString/BMPString instead of IA5String
- Invalid structure: violations of structural rules
- Discouraged field: e.g., CN in Subject or URI in SAN

# Quantifying Real-world Noncompliance

- Noncompliant Unicerts were issued by 505 issuer organizations, covering 78 CA owners in CCADB and 295 untrusted/unknown issuers.

Table: Top 10 issuer organization names by noncompliant unicerts.

Issuer OrgName	Trust status	Region	Noncompliant Unicerts	Recently Issued
Česká pošta...	●	CZ	22,939 (96.39%)	0
Symantec Corporation	○	US	18,092 (51.47%)	0
Dreamcommerce S.A.	●	PL	17,291 (44.83%)	0
DigiCert Inc	●	US	17,276 (3.40%)	40
Let's Encrypt	●	US	15,484 (0.06%)	7,091
StartCom Ltd.	○	IL	14,168 (72.97%)	0
COMODO CA Limited	●	GB	11,870 (0.25%)	0
ZeroSSL	●	AT	11,224 (2.53%)	4,094
Government of Korea	●	KR	10,416 (87.33%)	0
VeriSign, Inc.	●	US	7,513 (59.12%)	0
<b>Other</b>	-	-	103,008 (0.29%)	1,802
<b>Total</b>	-	-	249,281 (0.72%)	13,027

● publicly trusted ○ untrusted ● trusted in specific regions/scenarios

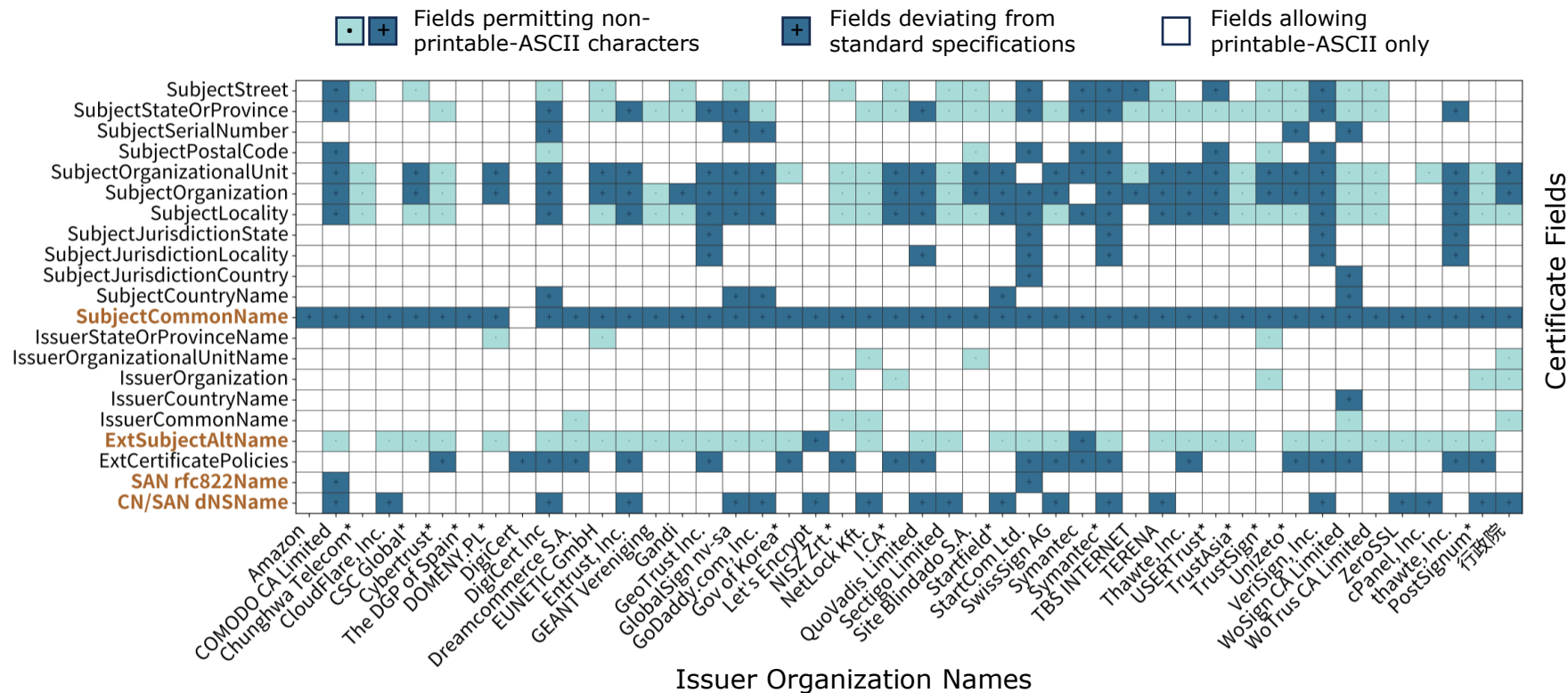
**Finding 1:** The problematic issuances involve both major **global CAs** (e.g., DigiCert, Let's Encrypt) and **regional providers**.

**Insight 1:** Some internationalization requirements **may not** be fully enforced by trusted CAs' certificate linting tools.

**Insight 2:** Automated domain validation workflows limiting field customization (e.g., Let's Encrypt allowing only DNSNames) **may** help reduce noncompliance.

# Troublesome Certificate Fields


- Issuers show **inconsistent** and **noncompliant** handling of certain non-ASCII fields.
- The encoding types, character ranges, or formats of **17 fields** were deviated from standard or baseline requirements.



21 fields permitting non-printable-ASCII chars or IDNs/IEAs/IRIs

# Case Study: The Problems in Certificate Fields

## I. Poor validation of DNSNames

 DNSNames in certs are critical for identifying peer entities.

- ❑ We identified 27K Unicerts with malformed IDNs:
  - i. cannot convert to Unicode
  - ii. include illegal characters (e.g., bidirectional controls) after Punycode decoding
- ❑ Example: **DNSNames in the SubjectAltNames extensions**

### Punycode (A-Label, ASCII Compatible)

```
xn--2ug.walesbonner.net  
xn--strandkrbe-kcapi.aosz.dev.bigfish.hu  
blog.xn--vpr-xda.statgrad.org
```

### Unicode (U-Label)

```
<0x200e>.walesbonner.net  
str<0x89><0x87>and<0x88>krbe.aosz.dev.bigfish.hu  
blog.<0xa0>vpr.statgrad.org
```



**Syntactically allowed under the current CA/B Baseline Requirements!**

### Deviation from multiple related standards?

- ✓ Valid P-Labels
- ✓ Resolvable via wildcard DNS: allowing domain validation
- ✗ Valid IDNs: U-Labels contain **DISALLOWED** characters

**RFC 5280:** Implementations **MUST** convert IDNs to ASCII Compatible Encoding (ACE) per **RFC3490** before storage in the dNSName.

**IDNA2003 suites** (RFC 3490 → RFC 3491 → RFC 3454)

**CA/B BR:** P-Labels not conforming to IDNA 2003 are allowed to support newer Unicode versions and updated IDNA standards.

**IDNA2008 suites** (RFC 5890 → RFC 5891 → RFC 5892)

U+200E, U+0080-U+009F, and U+00A0 are **DISALLOWED** characters

**CA side: Compliant**



**User agent side: Problematic**

# Case Study: The Problems in Certificate Fields

## II. Potential software defects or bugs.

- ❑ We found 117 Unicerts (issued by 20 organizations across 8 regions) with DN fields containing DEL (U+007F) characters.

Subject DN Attribute Value	Equivalent Value
Prepard[DEL][DEL]id Serc[DEL]vices	Prepaid Services
Xerox Cop[DEL]rporation   Xerox Cr[DEL]orporation	Xerox Corporation
Woodland h[DEL]Hills Data Center[DEL]*16	Woodland H

- **Observation:** Removing the text immediately preceding the DEL count reveals a meaningful, intended identity string.
- **Hypothesis:** This suggests an input-handling flaw where a delete keypress was recorded as a DEL character.

- ❑ We found 400 Unicerts (issued by IPS CA and Thawte Consulting) with DN fields containing special NUL (U+0000) characters.

Subject DN Attribute Value	Rendered String
[NUL]C[NUL]&[NUL]I[NUL]S	C&IS
[NUL]M[NUL]C[NUL]&[NUL]A	MC&A
[NUL]N[NUL]e[NUL]i[NUL]I[NUL] [NUL]H[NUL]a[NUL]r[NUL]v[NUL]e[NUL]y[NUL] [NUL]&[NUL] [NUL]A[NUL]s[NUL]s[NUL]o[NUL]c[NUL]i[NUL]a[NUL]t[NUL] [NUL]e[NUL]s	Neil Harvey & Associates

- **Observation:** The NUL characters were evenly inserted.
- **Hypothesis:** There might have encoding issues in issuance implementation.

# Case Study: The Problems in Certificate Fields

## III. Subject obfuscation: CAs allow variants without strict validation

- ❑ Allowing a broader Unicode range introduces vast variability in certificate fields.
- ❑ We discovered **six variation strategies** from the CT log data.

**Finding 1:** The same information may have multiple presentation formats.

**Finding 2:** There are Six subject variation strategies from the CT logs.

Different encodings of a French region name

Parsed Value	Unicode Presentation
Île-de-France	[U+00C3][U+0008]le-de-France
Île-de-France	[U+200E][U+00CE]le-de-france
ÎLE-DE-FRANCE	[U+00C3][U+0008]LE-DE-FRANCE
Ã?le-de-France	[U+00C3][U+0000]le-de-France
◆le-de-France	[U+FFFD]le-de-France
ile-de =-france	ile-de =[U+0008][U+0008]-france
île-de-France	[U+00EE]le-de-France

Variant Strategy	Variant Examples
Character case conversion	NOWOCZESNASTODOŁA.PL SP. Z O.O. nowoczesnaSTODOŁA.pl sp. z o.o.
Abbreviation variations	SKAT ELEKTRONIKS, OOO SKAT Elektroniks Ltd.
Addition of non-printable characters	PEDDY[U+00A0]SHIELD[U+00A0]... Peddy Shield ...
Use of different whitespace characters	株式会社[U+0020]中国銀行 株式会社[U+3000]中国銀行
Substitution of resembling characters	EDP-[U+002D] Energias de Portugal, S.A EDP-[U+2013] Energias de Portugal, SA
Replacement of illegal characters	St[U+FFFD]ri AG (TeletexString) Störi AG (UTF8String)

**Implication:** These variations increase the difficulty for **CAs** to verify identities and encodings, and for **relying parties** to correctly parse, compare, and store the information.



## ***Whether popular libraries respect declared encodings and enforce strict character checks?***

### ➤ **Methodology: Gray-box testing for TLS implementations.**

#### ❑ **Goal**

- Check whether TLS implementations properly **decode** and **validate** Unicert character ranges per normative requirements.

#### ❑ **Test cases: Constructed special test Unicerts using diverse:**

- Unicode blocks (e.g., C0 Controls)
- Encoding types (e.g., PrintableString, UTF8String, BMPString).

#### ❑ **Test scope**

- The certificate parsing APIs in 9 mainstream TLS libraries.

*OpenSSL, GnuTLS, Forge, PyOpenSSL, Cryptography, Golang Crypto, BouncyCastle, Java.security.cert, Node.js Crypto*

# Decoding and Character Handling Anomalies are Common

- We uncovered certificate field decoding or character handling anomalies in **ALL 9 tested TLS libraries**.
- Decoding Issues (Inconsistent Identity):
  - i. Incompatible Decoding:** Decoding values using non-standard-declared methods.  
*e.g., Forge decodes UTF8String using ISO-8859-1.*
  - ii. Over-Tolerant Decoding:** Decoding values using methods that permit a broader character range than defined by the standard encoding.  
*e.g., GnuTLS decodes PrintableString using UTF-8.*
  - iii. Modified Decoding:** Replacing undecodable bytes with substitute characters instead of rejecting them.
- Character Handling Issues (Validation Bypass):
  - i.** Every library exhibited standard violations in handling special characters (e.g., improper escaping, accepting characters beyond standard ranges).

# Case Study: Potential Security Implications

## Case I. Encoding-Decoding mismatches may cause Common Name forgery.

Encoding by CA: BMPString

```
\u7777\u777E\u6F72\u6163\u6C65\u2E63\u6F6D  
(CJK Unified Ideographs: 瞋 炯 湍 慣 汶 潭)
```

Decoding by library: ASCII

```
www.oracle.com
```

## Case II. Improper replacement of control character may cause CRL spoofing

URI in CRLDistributionPoints: (UTF8String)

```
http://ssl\u0001test.com
```

After replacement

```
http://ssl.test.com
```

## Case III. Allowing non-DNS characters in DNSNames can enable attribute embedding

Field value of a DNSName

```
DNSName="a.com DNS:b.com"
```

X.509-text representation

```
"DNS:a.com DNS:b.com"
```

String-based analyzers can misinterpret it as valid for two domain names.

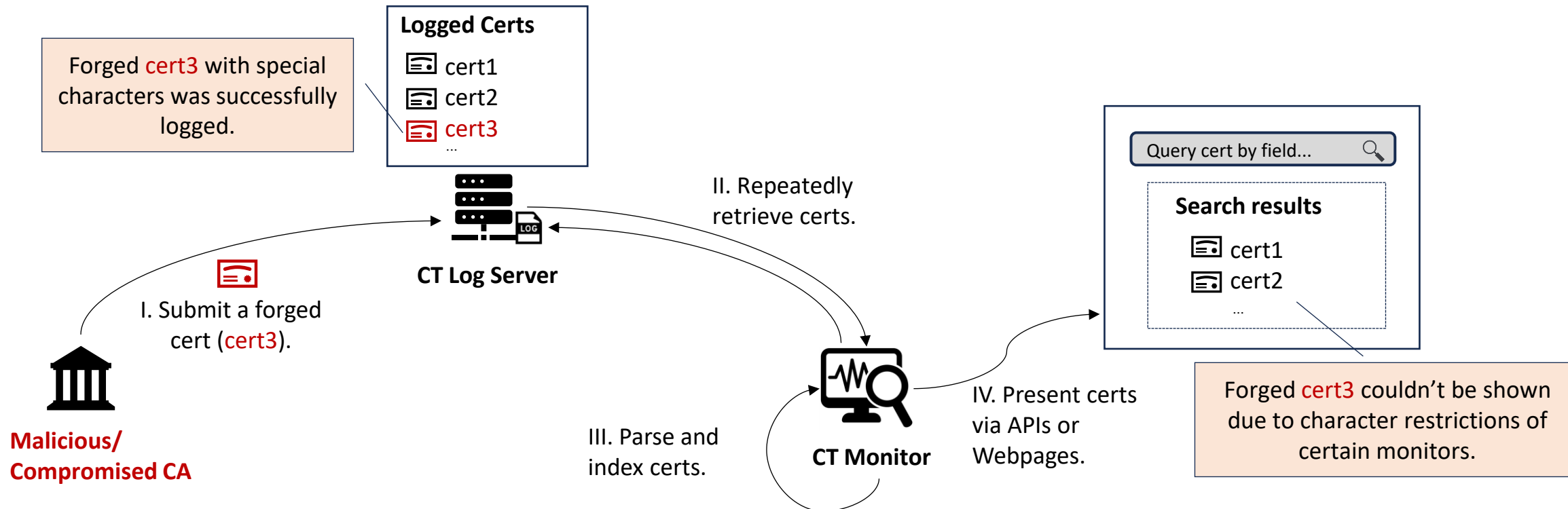
### Takeaways:

1. These threats are uncommon, as they require both flawed issuers and parsers.
2. We found certificates in CT logs that could cause misinterpretation but no evidence of exploitation.
3. Ongoing PKI practices may help mitigate these risks:  
*e.g., discouraging CN-based validation, using short-lived certificates to replace revocation checks*

# RQ3: The Real-World Threat Surfaces

## ❖ Misleading CT Monitoring

Maliciously malformed Unicerts can mislead Certificate Transparency (CT) monitors, allowing the concealment of specific forged certificates.



# RQ3: The Real-World Threat Surfaces

## ❖ User Spoofing

Crafted certificate fields can manipulate browser warning pages, potentially tricking users into trusting unverified sites, though practical exploitation requires **stringent conditions** and would have **limited impact**.



### Your connection is not private

Attackers might be trying to steal your information from **www.example.com** (for example, passwords, messages, or credit cards). [Learn more about this warning](#)

NET::ERR\_CERT\_COMMON\_NAME\_INVALID

Subject: **www.paypal.com**

Issuer: Self-signed Root CA

Expires on: Dec 24, 2024

Current date: Sep 25, 2024

Subject: **www.<0x202e>lapyap<0x202c>.com**

Advanced

Back to safety

Websites prove their identity via certificates. Firefox does not trust this site because it uses a certificate that is not valid for **www.example.com**. The certificate is only valid for port 8443. But **they're the same site**. You can continue or try: <http://www.example.com>.

Error code: [SSL\\_ERROR\\_BAD\\_CERT\\_DOMAIN](#)

[View Certificate](#)

The highlighted string is parsed from the **DNSName** within the **SubjectAltName** field.

Go Back (Recommended)

Accept the Risk and Continue

**Chrome case:** the warning page renders bidirectional characters in the Subject CommonName field.

**Firefox case:** the page displays misleading information derived from a malformed SubjectAltName field.

# Conclusion, Mitigation, and Contributions

## ➤ **Summary**

- Achieving a truly internationalized PKI is currently challenging due to systemic issuance noncompliance and universal parsing flaws.

## ➤ **Key takeaways and current consensus**

1. Building an internationalized PKI requires collaboration among CAs, developers, and standards bodies.
2. Unicode integration spans evolving standards, while current CA/B BRs lag behind updates (e.g., IDNA 2008).
3. Correct ASN.1 parsing in Unicerts is as essential as compliant issuance.

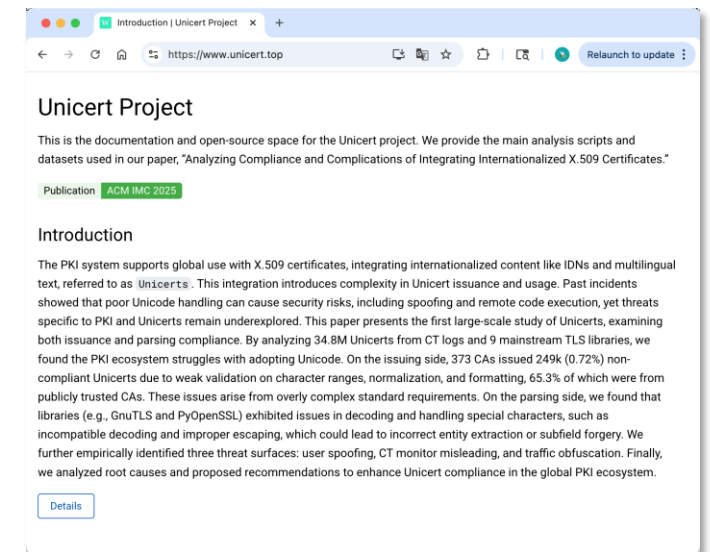
# Conclusion, Mitigation, and Contributions

## ➤ Our Contributions:

- First systematic study on **Unicert issuance and parsing** compliance.
- Identification and large-scale prevalence measurement of 3 types of issuance noncompliance.
- Empirical discovery of decoding and character handling anomalies in all 9 tested TLS libraries.

## ➤ Mitigation

- We responsibly disclosed the issues and provided tools/recommendations to affected CAs and TLS library teams; coordinated on fixes.
- We released our testing tools for future work and more discussions with the community



Project website: <https://www.unicert.top>

# Thank you!

## Q&A

Presenter: Mingming Zhang

**Contacts:**

[zhangmm717@gmail.com](mailto:zhangmm717@gmail.com)

**Paper available at:**

<https://www.unicert.top>

<https://doi.org/10.1145/3730567.3764483>