



RUB

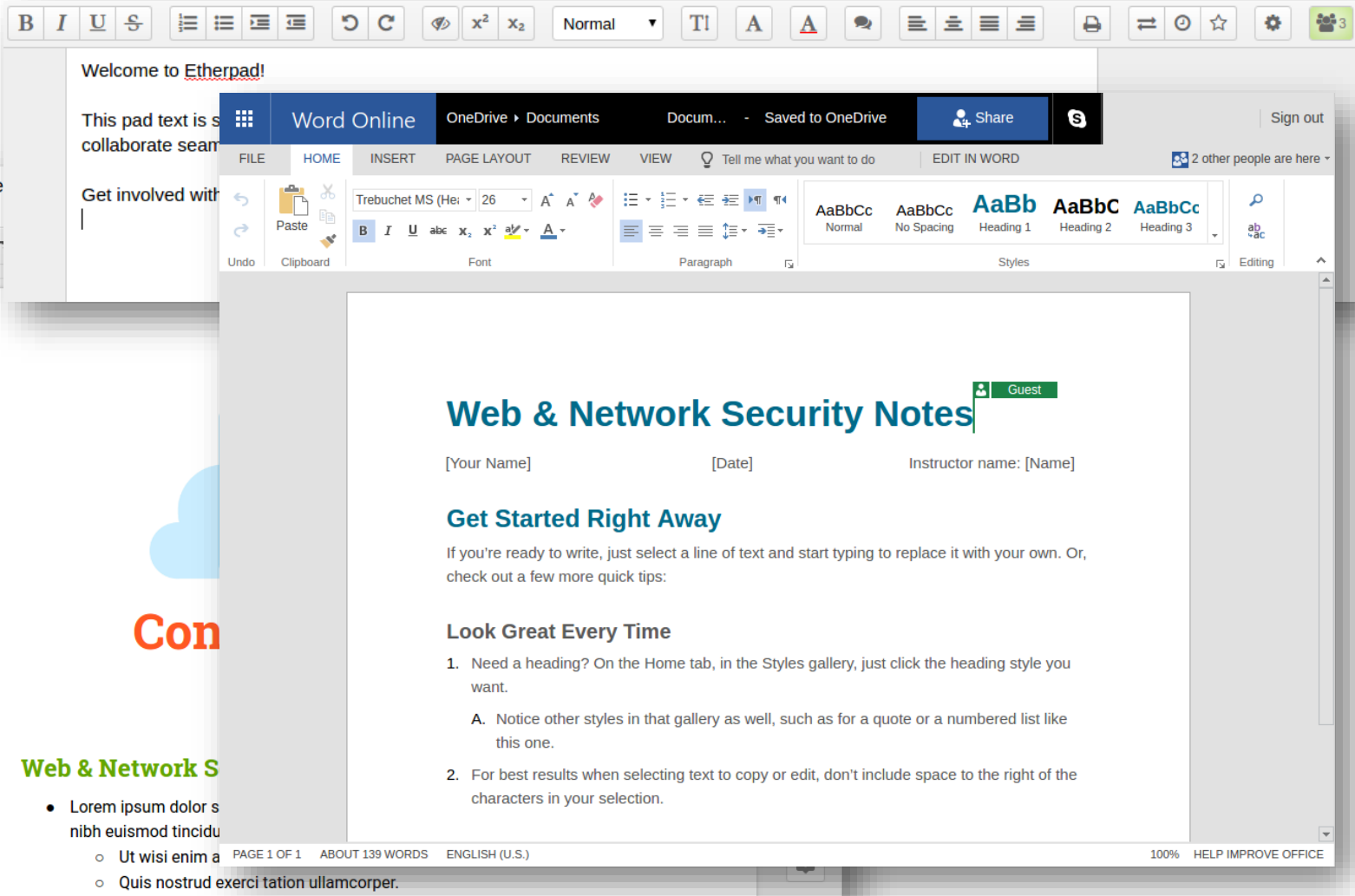
RUHR-UNIVERSITÄT BOCHUM

# **SECRET: On the Feasibility of a Secure, Efficient, and Collaborative Real-Time Web Editor**

**Dennis Felsch, Christian Mainka,**  
**Vladislav Mladenov, Jörg Schwenk**

hg **i** NDS

# Real-Time Web Editing Tools



# Operational Transforms (OT)

- Maintain a consistent view on a document
- Automatically resolve editing conflicts
- Whole area of research on its own

# Motivation

- Established tools do not apply (cryptographic) protection to documents
- Previous academic proposals with encryption either
  - Require large overheads
  - Are not real-time collaborative,
  - Require browser extensions, or
  - Do not take structure into account
- Is it feasible to have all these properties?

# SECRET

- First **S**ecure, **E**fficient, and **C**ollaborative **R**eal-time **E**ditor
- SECRET is the first collaboration tool with
  1. encryption of whole documents or arbitrary sub-parts,
  2. novel combination of tree-based OT and structure preserving encryption,
  3. only a modern browser without any extra software installation or browser extension required

# Building Blocks

- ShareJS
  - JavaScript Middleware with OT algorithms
- State-of-the-Art Web Technologies
  - WebSockets for Asynchronous Messaging
  - W3C Web Cryptography API for AES-128 in Galois Counter Mode (GCM)
  - PostMessage API
- XML Encryption
  - Structure Preserving Encryption

# XML Encryption

```
<PaymentInfo>
  <Name>John Smith</Name>
  <CreditCard Limit="5,000" Currency="USD">
    <Number>1234 5678 2580 1595</Number>
    <Issuer>Example Bank</Issuer>
    <Expiration>04/17</Expiration>
  </CreditCard>
</PaymentInfo>
```

```
<PaymentInfo>
  <Name>John Smith</Name>
  <EncryptedData
    Type="http://www.w3.org/2001/04/xmlenc#Element"
    xmlns="http://www.w3.org/2001/04/xmlenc#">
    <CipherData>
      <CipherValue>184797A8C2FE977DEFA10A7FE540A0D0</CipherValue>
    </CipherData>
  </EncryptedData>
</PaymentInfo>
```

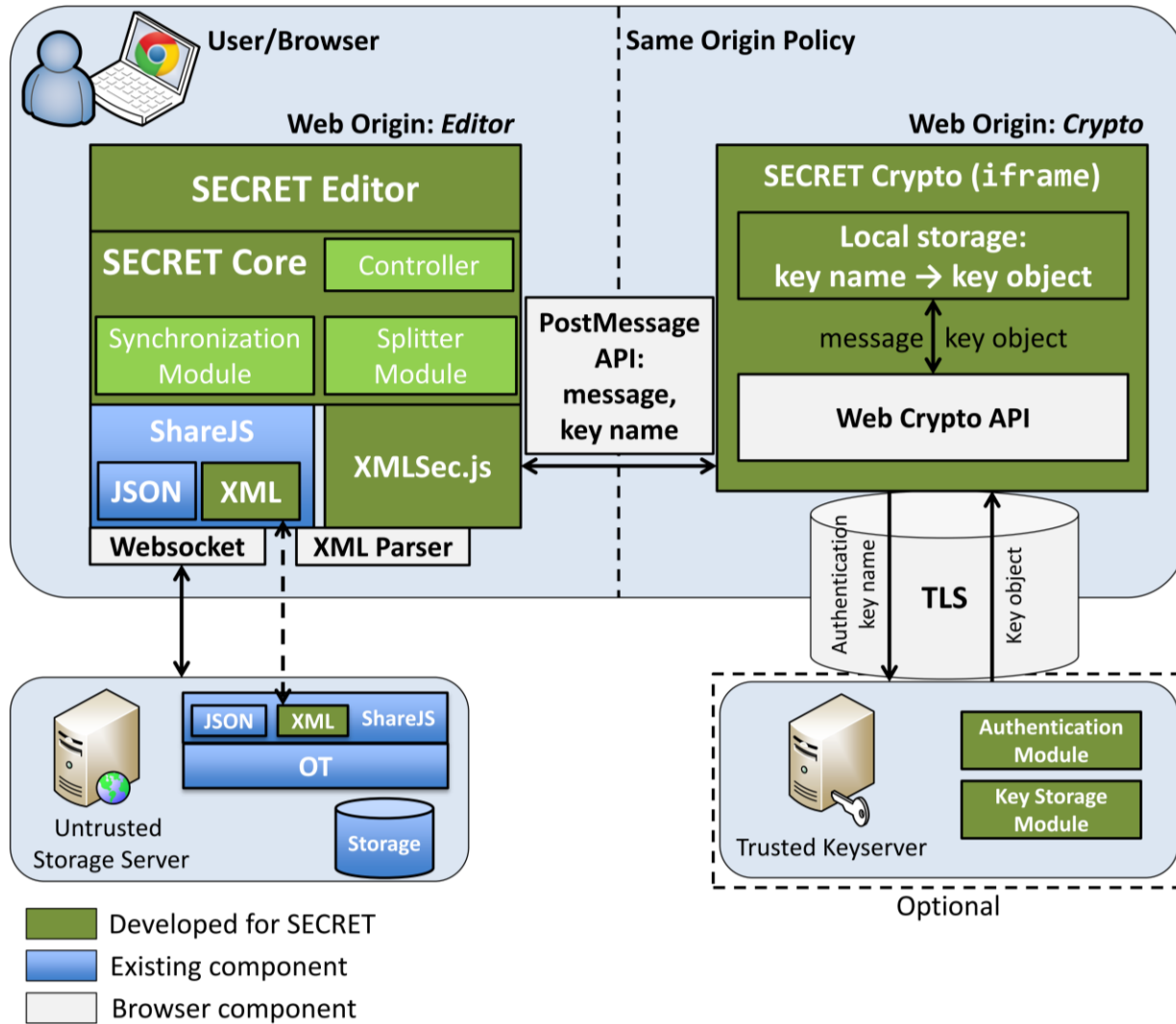


# Implementation Challenges

- ShareJS does not support XML
  - Solution: Implemented OT for XML documents as an extension of ShareJS
- Browsers do not support XML Encryption
  - Solution: Implemented a JavaScript library to encrypt, decrypt, sign, or verify documents
- WebCrypto API does not handle long-lived, persistent keys
  - Solution: Store them on an key-server or derive them from a password



# Architecture Overview



# SECRET

SECRET: Secure, Efficient, and Collaborative Real-Time Web Editor

Home | SECRET Demo | **ONLINE**

Split Size: 4

**Part 1 - Encrypted**  
abcdef

**Part 2 - Encrypted**  
xyzijkopq

**Part 3 - Plaintext**  
123

Show me the ciphertext

Please use the following credentials:

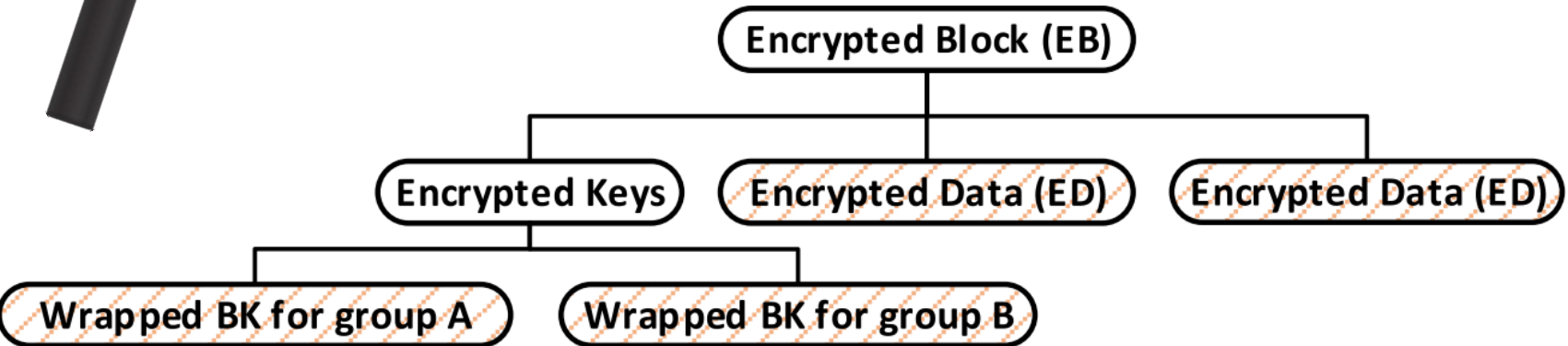
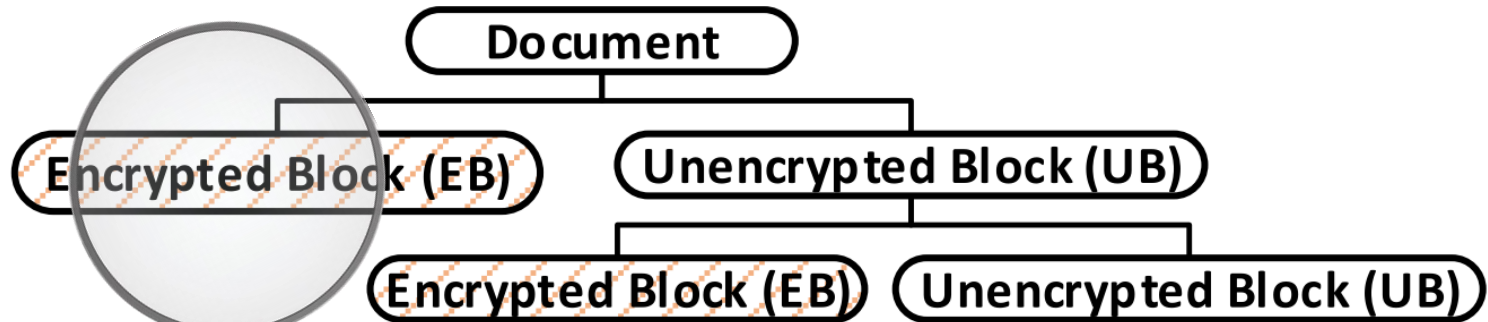
- Username: john.doe
- Password: 12345

Username:

Password:

Log on

# Documents



# Screenshot Ciphertext

The screenshot shows a web browser window with the address bar displaying `ec2-54-234-131-44.compute-1.amazonaws.com:8080/secret_plain`. The page title is "SECRET: Secure, Efficient, and Collaborative Real-Time Web Editor". Below the title is a navigation bar with "Home", "SECRET Demo", and "ONLINE" (highlighted in green). The main content area is titled "Live XML of the document:" and contains a preformatted XML document. The XML document is a mix of plain text and encrypted parts. The visible XML is as follows:

```
<document><encryptedpart id="e9a"><ed id="_obpub15u">
<cv>axmL07qRwaAtWraJPof9eIY/cVmlQDZtdd2YDTwCt7FYJcGGmHypU13aumkE4ONQ</cv></ed><ed
id="_38gcqnc7"><cv>Er89ACI+nCp3Qtm2OIMJHf646rs8m906syAStQsAY2Di59gEozzMFD+5QT+Hw98q</cv>
</ed><ek id="_wt80vryj"><cv>n6g03qvWtPT7NF+3928uZzYhmrPfkH3N</cv><ckn>_p3f35p7u-1</ckn></ek><ek
id="_isx46dwt"><cv>a/lKAetG7/pZjsF6eXGMQvH9GNqotY9n</cv><ckn>_p3f35p7u-2</ckn></ek>
</encryptedpart><encryptedpart id="b7d"><ed id="_fzhy53xf">
<cv>y1mu72nEf3YNKgZ16YnUba9p7VLulAhHO9V8c1sSKXq6GV5xVLjQV4ewwilTLKGI</cv></ed><ed
id="_m6olxyhx"><cv>x94HsVqralRT9oaE1sB/Gve+Hm81trKS3koX+YKhez4hb8hKA74z2+I6LQS8+Eil</cv>
</ed><ed id="_oibwcyvq">
<cv>yeFPSTbbtvg96ceT8ClmKEDLoRVEY1kaGr+lxwHuqpGMyT8zxpZ0PD6MDD8x7nyK</cv></ed><ek
id="_5c7rqocv"><cv>TwYfythVQURIKRXsya8iHbD09SSqJaoY</cv><ckn>_8kkf12m0-2</ckn></ek>
</encryptedpart><part id="f7c"><span>123</span></part></document>
```

Below the XML, there is a note: "Everyone editing [this page](#) modifies the XML you can see here. Edit it yourself to see the effect."

# Splitting into Encrypted Data Chunks

- Large encrypted blocks
  - ⇒ Updates are inefficient
  - ⇒ Splitting is necessary
- Small encrypted blocks
  - ⇒ Large XML overhead

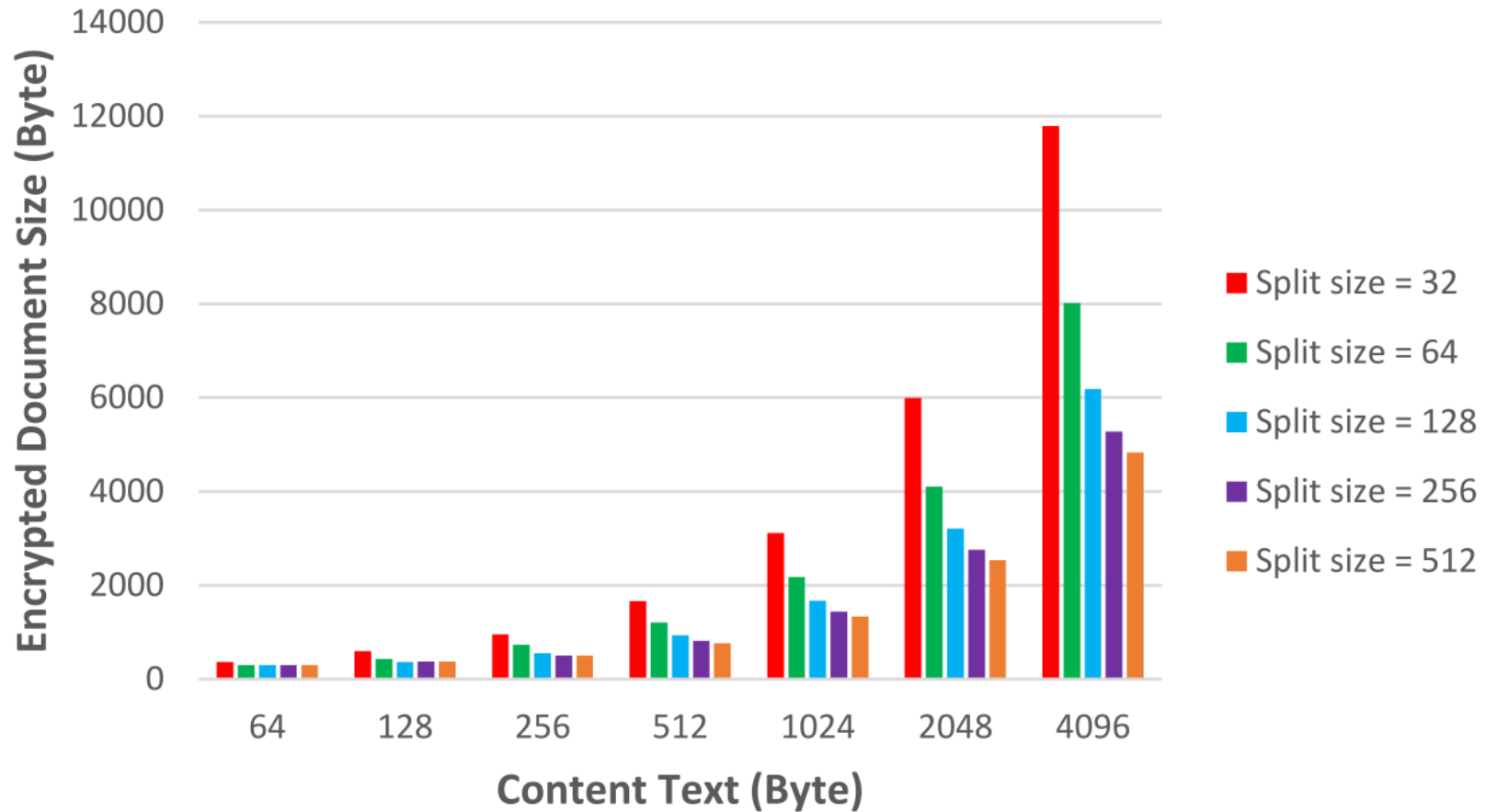
```
<div>  
  <span>Hello</span>  
  <span>World</span>  
</div>
```

- Q: What is the optimal split size?

# Evaluation

- Google Chrome 50 with Selenium
- Simulated typing at 200 key strokes / min
- Measured storage and network overhead

# Evaluation Storage





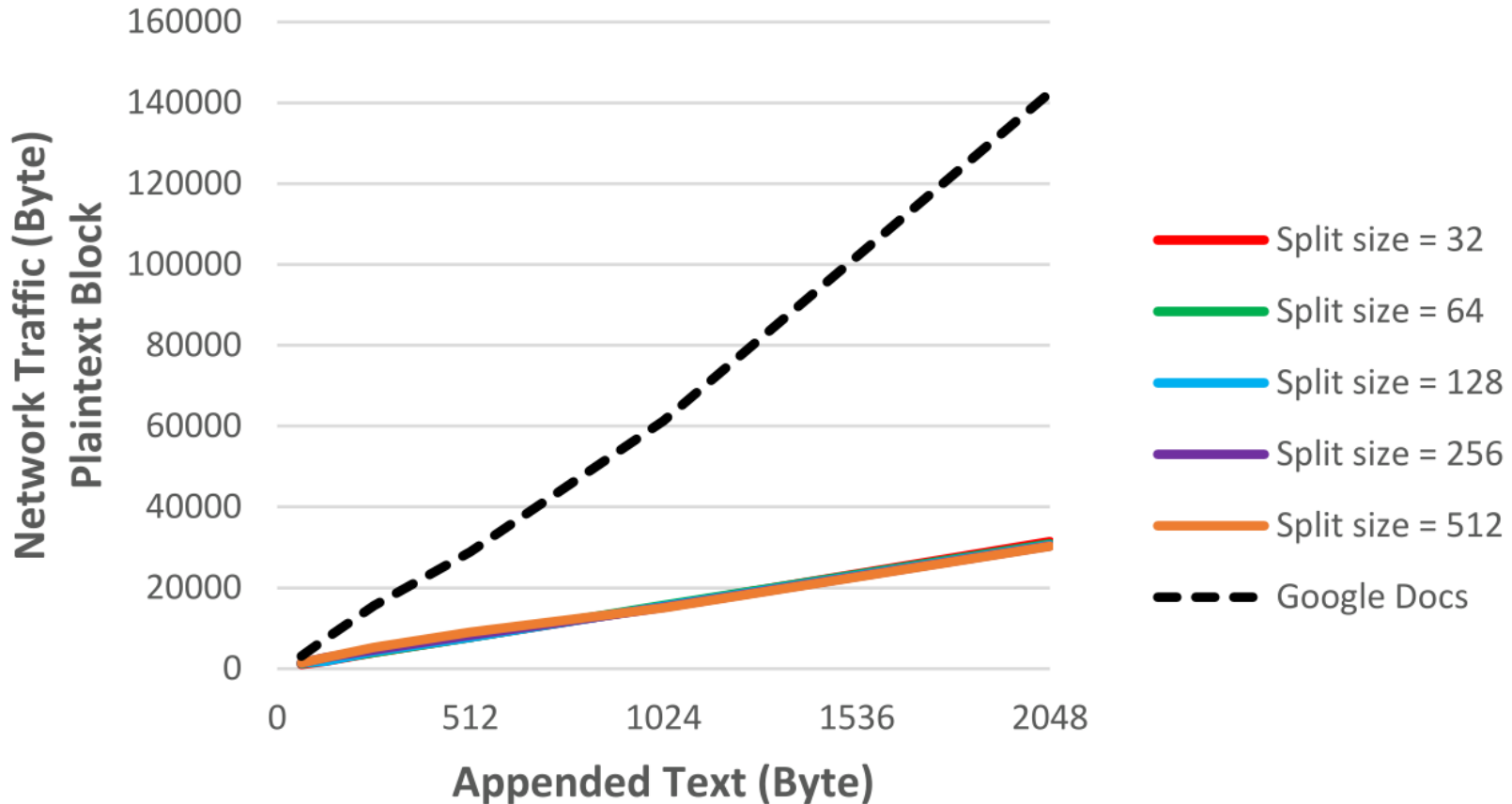
# Evaluation Storage

<b>Split size</b>	32	64	128	256	512
<b>Storage expansion</b>	3.50	2.46	1.92	1.66	1.53

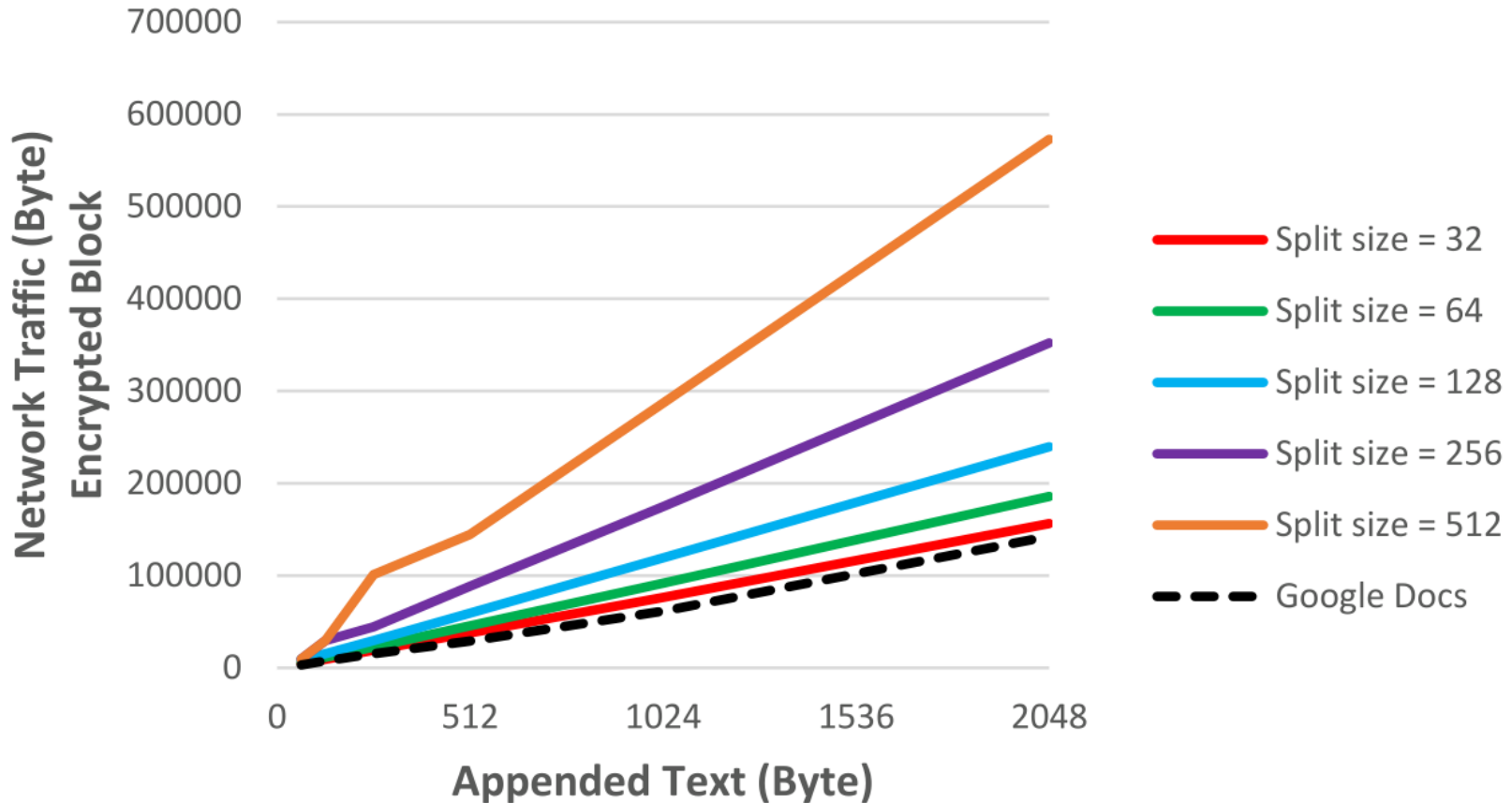
Table 2: Ciphertext expansion of a 4096 byte document.

- Numbers look high
- In fact, they are far below the numbers in related work
- Best results before ours: 3.75 – 4.82

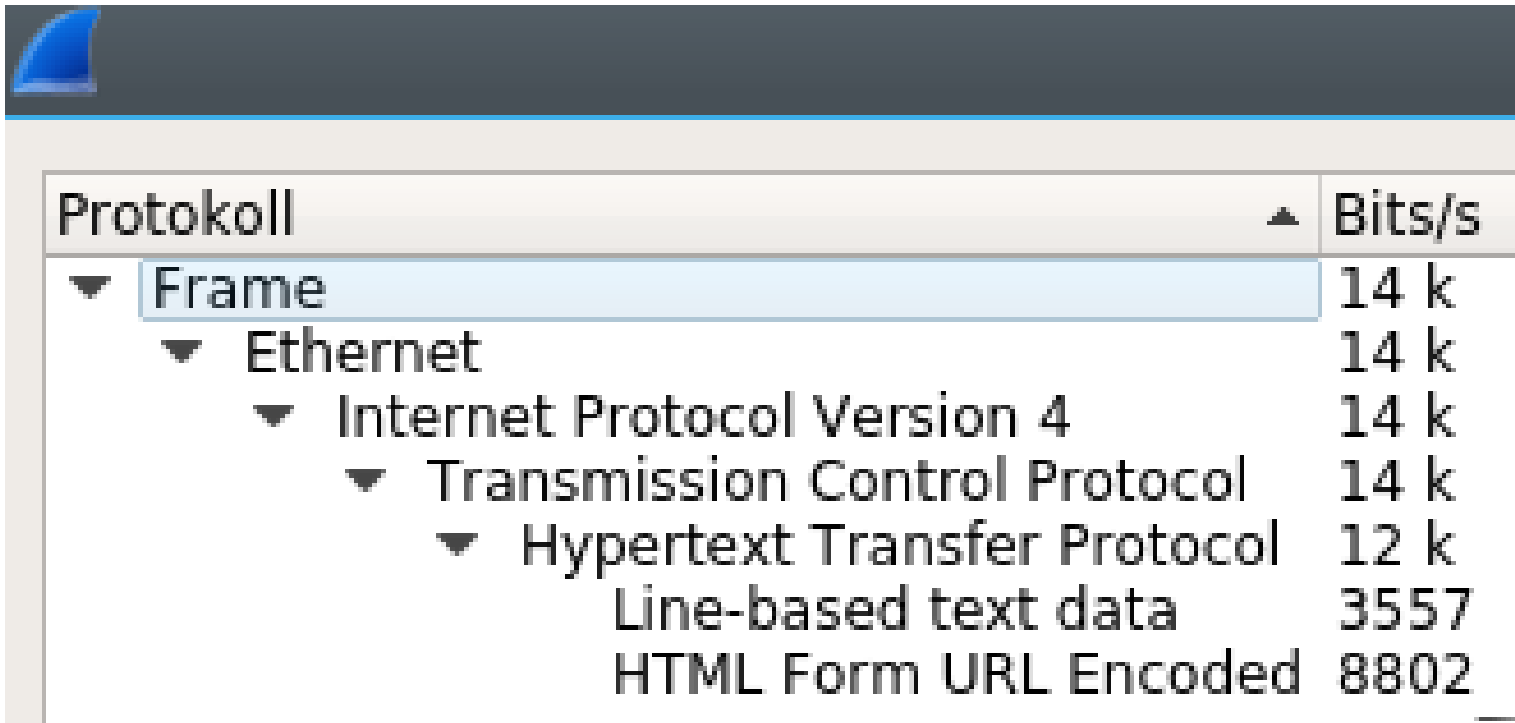
# Evaluation Network



# Evaluation Network



# Bandwidth Requirements



Protokoll	Bits/s
▼ Frame	14 k
▼ Ethernet	14 k
▼ Internet Protocol Version 4	14 k
▼ Transmission Control Protocol	14 k
▼ Hypertext Transfer Protocol	12 k
Line-based text data	3557
HTML Form URL Encoded	8802

Figure 7: Screenshot from *Wireshark* measuring the required bandwidth for SECRET at 200 key strokes per minute with a split size of 128 bytes.

# Conclusion & Outlook

- A Secure, Efficient, and Collaborative Real-Time Web Editor is feasible
- No need for large overheads when using Structure Preserving Encryption
- GUI and editing features can be improved
- How about full-fledged office documents?
- SECRET's code is on GitHub:  
<https://github.com/RUB-NDS/SECRET/>

# Questions?



Dennis Felsch  
Horst Görtz Institute for IT-Security  
Chair for Network and Data Security  
Ruhr-University Bochum

dennis.felsch@rub.de

Twitter: @dfelsch

NDS-Blog:  
<http://web-in-security.blogspot.de>