Cybersecurity

Wintersemester 2025/2026

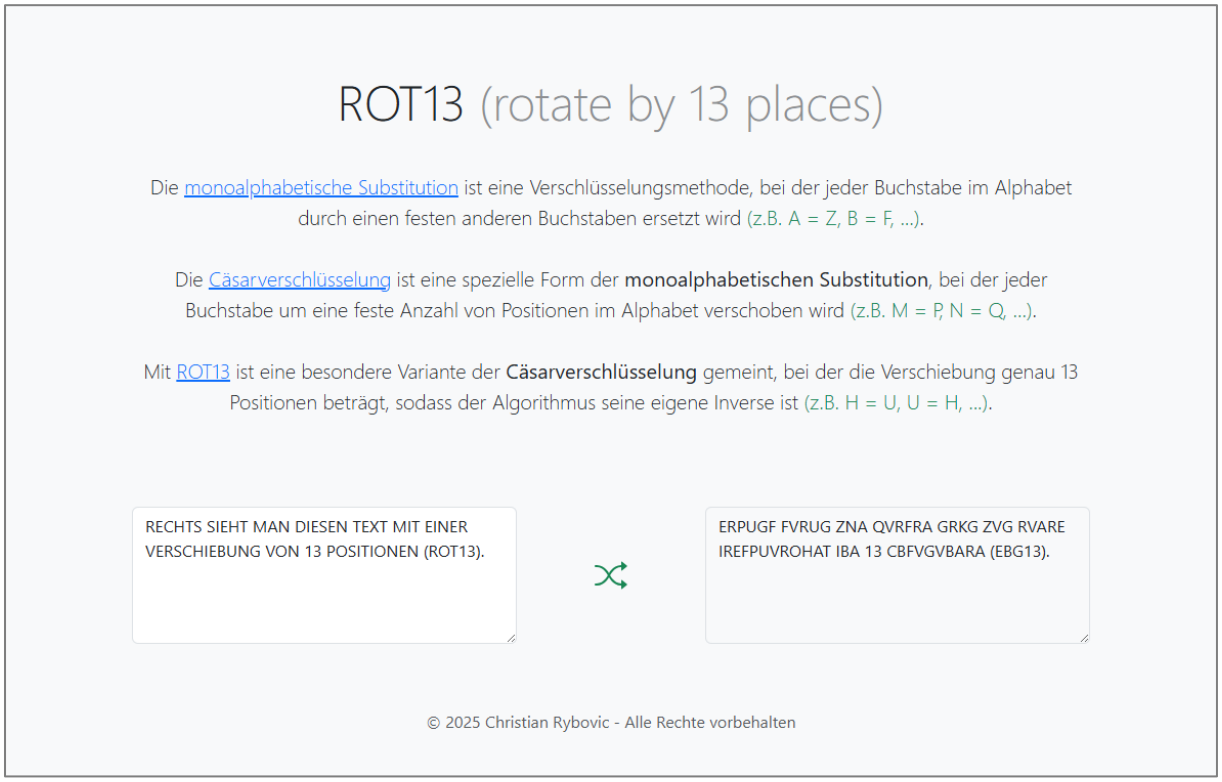
Christian Rybovic
L-TIP-24-Do-a

«Cybersecurity umfasst Massnahmen zur Verhinderung, Erkennung und Abwehr von Cyberbedrohungen, um Vertraulichkeit, Integrität und Verfügbarkeit von Informationen zu gewährleisten.»

Inhalt

Kompetenznachweis 1 vom 30.10.25

Zum besseren Verständnis wie eine symmetrische Verschlüsselung nicht nur in der Theorie, sondern auch in der Praxis technisch umgesetzt wird, wurde eine Webseite mit einer einfachen ROT13 Substitution erstellt. Unter https://neo.christianrybovic.ch/rot13-app hat man die Möglichkeit, einen chiffrierten oder nicht-chiffrierten Text einzugeben, welcher dann über den ROT13-Algorithmus umgewandelt wird.



Screenshot der Webseite mit einem Beispiel

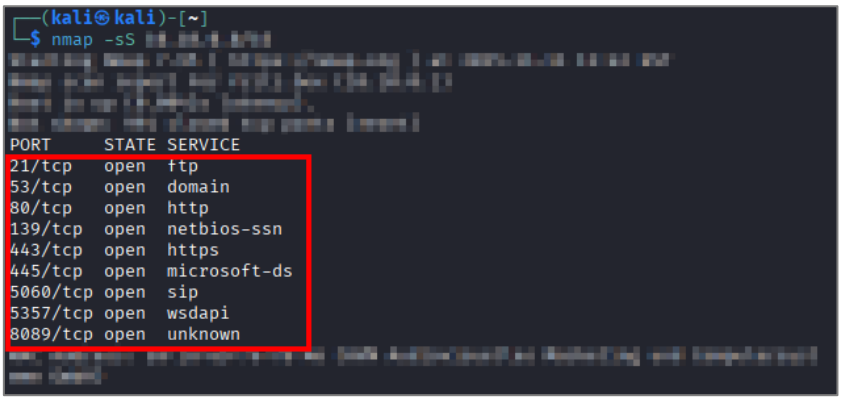
Bei ROT13 handelt es sich um eine Abwandlung der Cäsarverschlüsselung, welche wiederum eine monoalphabetische Substitution ist. Da der Algorithmus mit sich selbst Invers ist, muss nicht zwischen Chiffrierung und Dechiffrierung unterschieden werden - nach jedem zweiten Durchlauf erhält man wieder die ursprüngliche Eingabe.

Die Oberfläche der Webseite wurde mithilfe des Bootstrap-Frameworks erstellt, die Umwandlung des Textes erfolgt per JavaScript. Der Quellcode kann in jedem gängigen Browser betrachtet werden.

Kompetenznachweis 2 vom 06.11.25

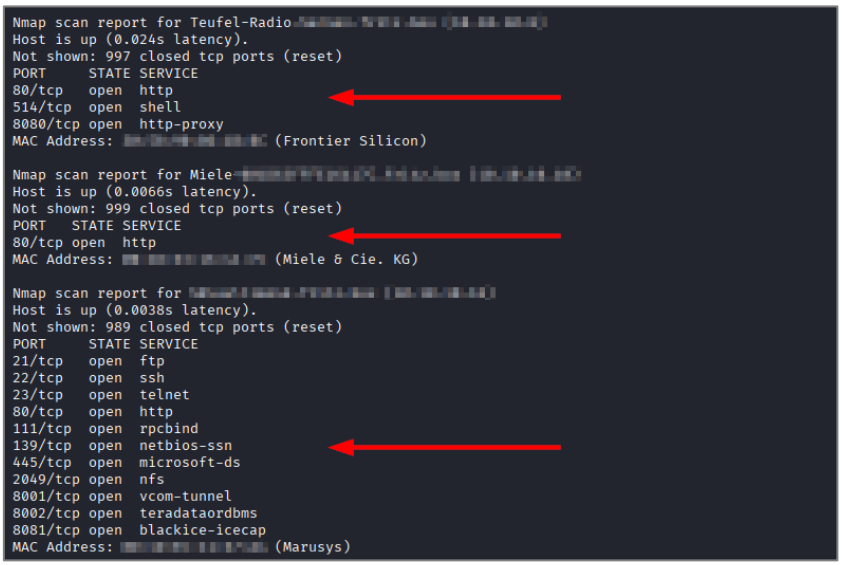
Mit Kali Linux bereits mitgeliefert wird das Programm **nmap**. Das Programm kann direkt über das Terminal ausgeführt werden. Je nach Einstellung kann damit sehr schnell ein Netzwerk gescannt und wichtige Informationen gesammelt werden.

Ohne einen Suchlauf mit **nmap** erwartet man ein paar wenige Geräte im Netzwerk. Man zählt vielleicht die Notebooks und Workstations auf, evt. denkt man noch an Drucker und Smartphones. Viele weitere Geräte werden vergessen, sodass ein einfacher Suchlauf, welcher nur ein paar Sekunden dauert, oft zu grossem Erstaunen führt. In der Testumgebung konnte so 13 Geräte gefunden werden, wobei die meisten Geräte auch mehrere Ports geöffnet haben.



Beispiel eines Gerätes mit mehreren aktiven Ports

Neben den gängigsten Ports wie HTTP, HTTPS und FTP, lassen sich auch Ports finden, welche man eigentlich nicht erwartet hätte. So haben einfache Satreceiver für den Fernseher zum Beispiel den TELNET- und SSH-Port, eine Waschmaschine den HTTP-Port oder ein Internetradio den SHELL-Port geöffnet.



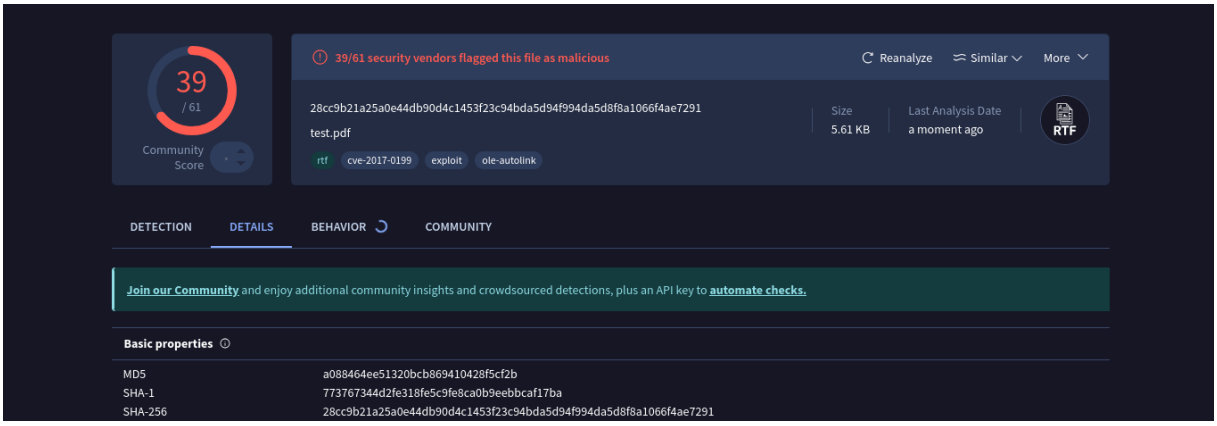
Ports von einem Satreceiver, einer Waschmaschine und einem Internetradio

Kompetenznachweis 3 vom 13.11.25

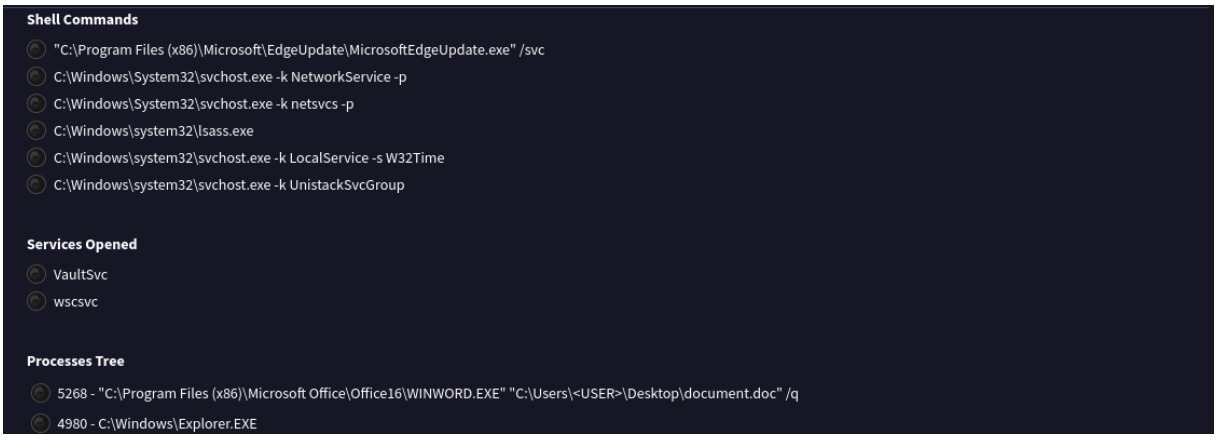
Mit Metasploit kann ein Angriff mit einem Payload folgendermassen ablaufen:

1. Metasploit-Konsole mit dem Befehl «**msfconsole**» starten.
2. Nun kann mit dem Befehl «**search pdf**» nach Schwachstellen im Zusammenhang mit PDF gesucht werden. Metasploit listet anschliessend verschiedene Module auf. Typische Kategorien sind beispielsweise:
    a. Exploits für ältere Adobe-Reader-Versionen
    b. Embedding-Techniken (z. B. JavaScript im PDF)
    c. File-format Exploits
3. Der gewünschte Exploit kann dann mit dem Befehl «**use exploit/windows/fileformat/office_word_hta**» ausgewählt werden.
4. Mit «**set payload generic/debug_trap**» wird der Payload angegeben.
5. Anschliessend noch mit «**set FILENAME test.pdf**» die Datei definieren und mit «**exploit**» erstellen.

Auf einem ungepatchten System würde der Payload ohne weiteres geladen werden. Wie der nachfolgende Screenshot zeigt, wird die Datei von den meisten AV-Programmen erfolgreich als Schädlich eingestuft. Z.B. BitDefender klassifiziert die Datei als «**Trojan.Script.746654**».



Die Verhaltensanalyse zeigt zudem weitere interessante Informationen an. Dazu gehört z.B. welche Dateien angelegt oder gelöscht, welche Registry-Schlüssel modifiziert, oder welche Services geöffnet werden:



Kompetenznachweis 4 vom 20.11.25

Cross-Site Scripting (XSS) gehört zu den häufigsten und gefährlichsten Sicherheitslücken in Webanwendungen. Die Schwachstelle entsteht, wenn Benutzerinnen Eingaben in eine Webseite einbringen können, die später ohne ausreichende Prüfung oder Filterung wieder im Browser anderer Nutzerinnen ausgegeben werden. Angreifer nutzen dies aus, um schädlichen JavaScript-Code in eine Webseite einzuschleusen. Es gibt drei Hauptformen von XSS:

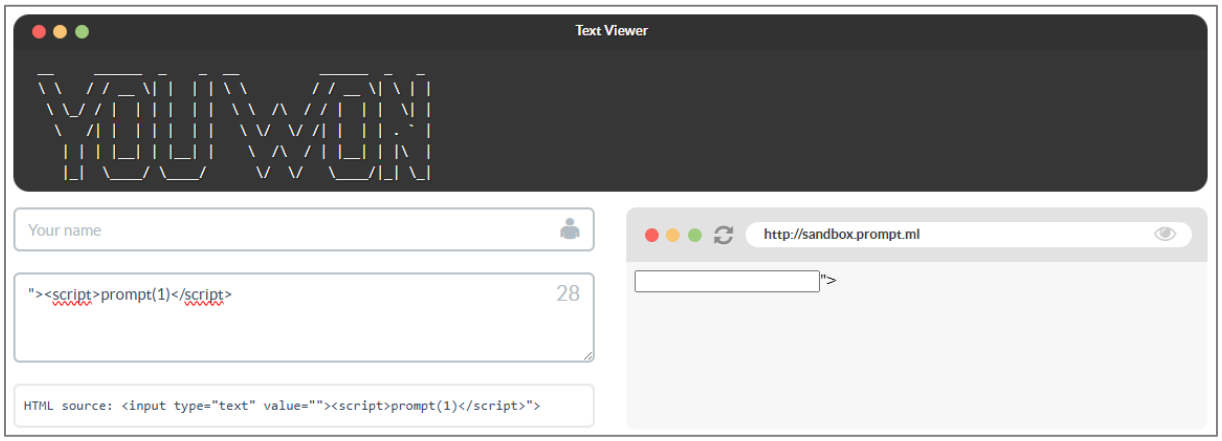
Stored XSS (Persistent XSS): Hier wird der schädliche Code dauerhaft z.B. in einer Datenbank gespeichert und bei jedem Aufruf einer bestimmten Seite an die User ausgeliefert.

Reflected XSS: Der Angriffscode wird über eine URL oder ein Formular an den Server geschickt und direkt in der Antwort reflektiert. Der User muss meist auf einen präparierten Link klicken.

DOM-basiertes XSS: Beim DOM-XSS findet die Manipulation vollständig im Browser statt. JavaScript auf der Seite verarbeitet Benutzereingaben unsicher und führt so den Angriff aus, ohne dass der Server betroffen ist.

Auf der Webseite https://prompt.ml gibt es verschiedene Beispiele um XSS besser zu verstehen. Die Seite zeigt im oberen Bereich den Code an, welcher verwendet wird. Auf der linken Seite gibt es Felder für die Eingabe und rechts sieht man den Output.

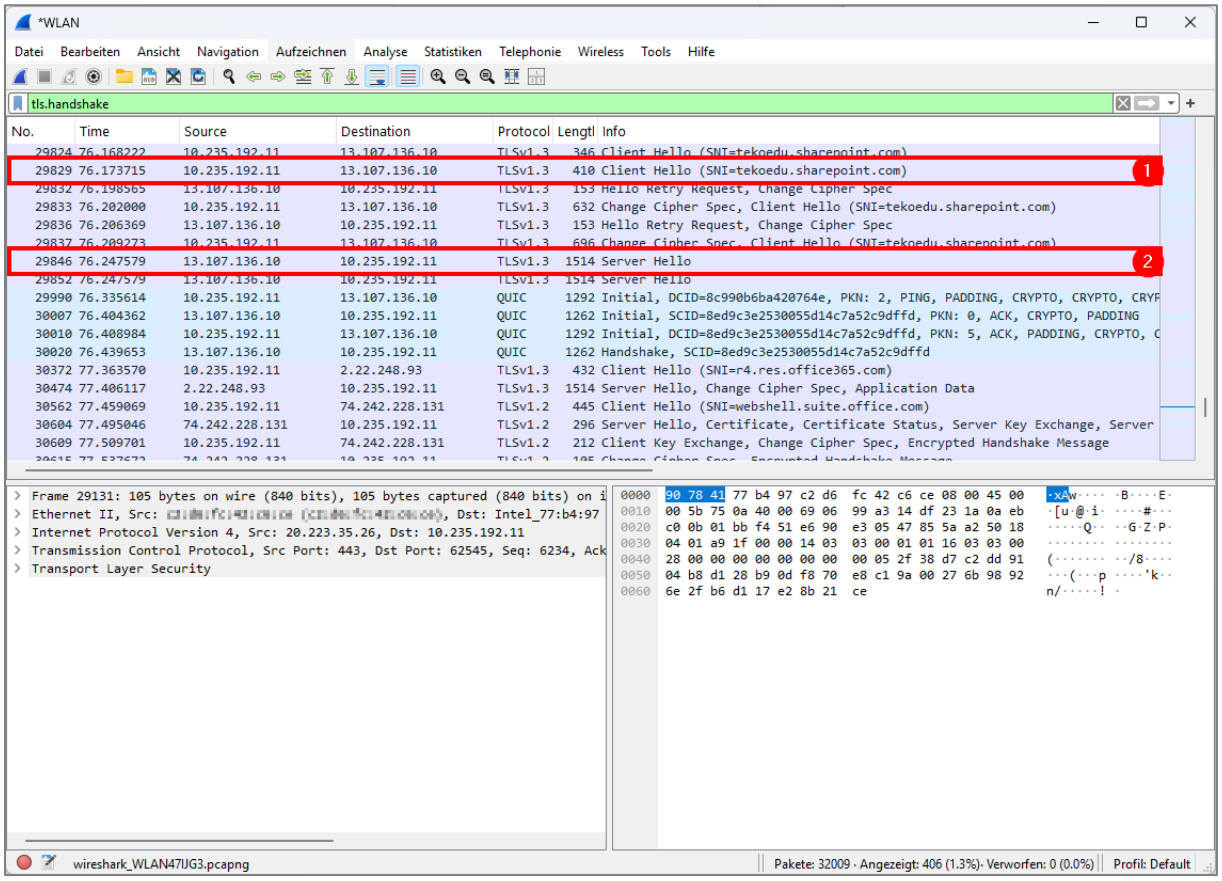
Ziel ist es, die Webseite dazu zu bringen, «prompt(1)» auszuführen. Das erste Beispiel ist recht rudimentär. Hier muss einfach der HTML-Code über das Input-Feld zuerst abgeschlossen werden und schon kann man den Script-Code einfügen:



Kompetenznachweis 5 vom 27.11.25

Um Man-in-the-Middle-Angriffen im Web vorzubeugen, werden Verbindungen durch TLS abgesichert. Dies ist die Grundlage für HTTPS. Dies erschwert es einem Angreifer zwar nicht, die übertragenen Daten abzufangen, aber er kann diese nicht so einfach mitlesen oder manipulieren. Als weiterer Vorteil kann von den beteiligten Parteien die Identität mit dem Zertifikat überprüft werden.

In einer Wireshark-Aufzeichnung kann man dann mit dem Filter "tls.handshake" die Handshakes anzeigen lassen. Im ersten Paket sendet der Client eine Liste unterstützter Cipher Suites, Protokollversionen sowie eine Zufallszahl (**1**). Anschliessend antwortet der Server mit seiner eigenen Zufallszahl sowie dem gewählten Verschlüsselungsverfahren (**2**).



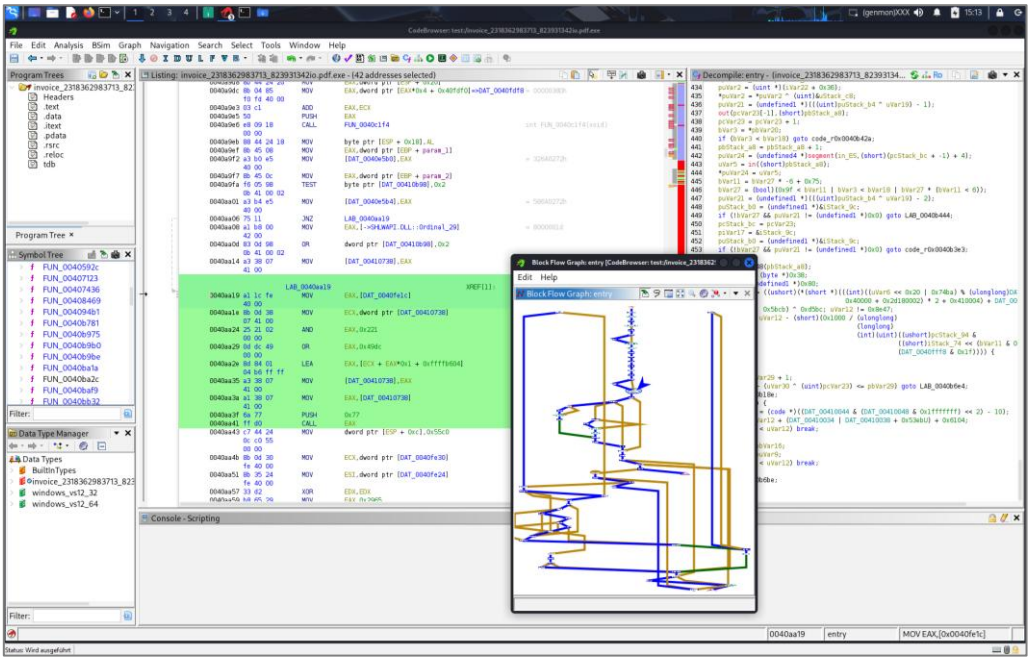
TLS-Handshake in Wireshark

Im Beispiel vom aufgenommenen Screenshot wurde mit dem SharePoint der Schule verbunden. Die ersichtliche SNI (Server Name Indication) zeigt, dass hier der korrekte SharePoint angesprochen wird (tekoedu.sharepoint.com).

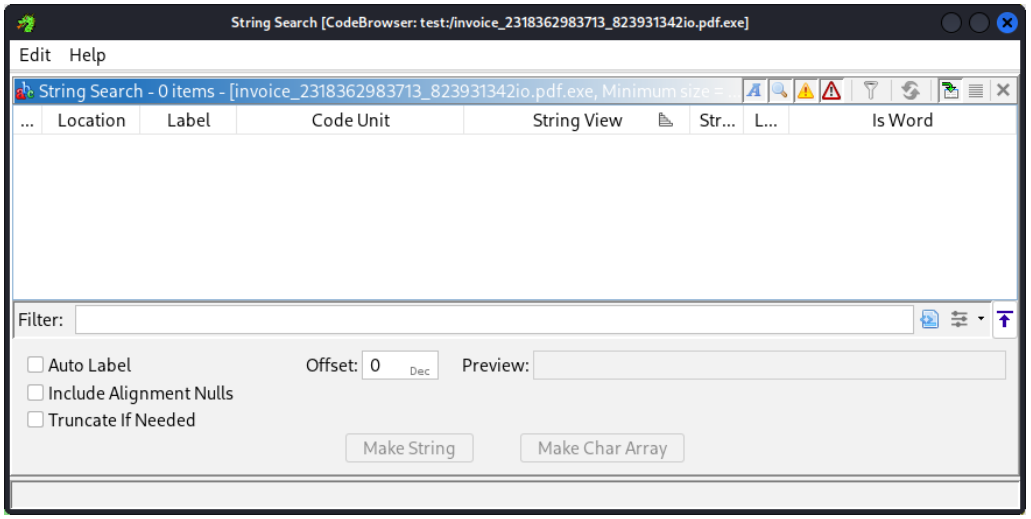
Da auf dem Client ständig irgendwelche Programme Verbindungen aufbauen, füllt sich so eine Aufzeichnung mit Wireshark sehr schnell. Deshalb ist es wichtig, mit Filtern zu arbeiten und ungefähr zu wissen, nach was man in der Aufzeichnung überhaupt suchen möchte.

Kompetenznachweis 6 vom 04.12.25

In diesem Kompetenznachweis geht es um die Analyse von Malware. Bei theZoo handelt es sich um ein Malware-Repository auf GitHub. Da es sich hierbei um echte Malware handelt, welche analysiert wird, sollte zwingend eine virtuelle Maschine und eine abgesicherte Umgebung verwendet werden. Zu empfehlen ist natürlich Kali Linux, weil dieses bereits mit sehr nützlichen Tools für die Analyse ausgestattet ist. Auf dem nachfolgenden Screenshot wurde so eine Malware mit Ghidra geöffnet. Auf diesem sieht man z.B. den Code in Assambly und Pseudocode sowie den Block Flow Graph.

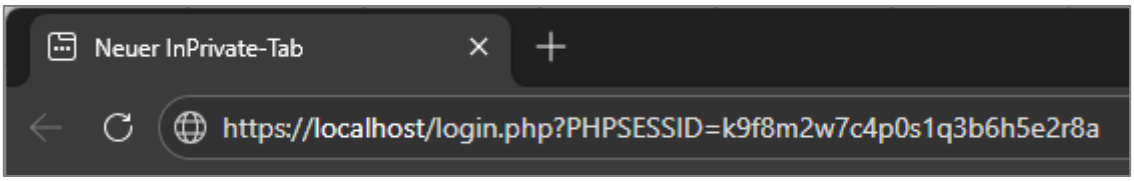


Bei einer neuen Analyse ist es immer praktisch, beispielsweis nach Zeichenketten über Ghidra zu suchen. Normalerweise wird immer da und dort ein String gefunden (selbst wenn sie keinen Sinn ergeben). Aber bei dieser Malware konnte Ghidra überhaupt nichts finden. Der Grund können natürlich auch Verschleierungstechniken von den Entwicklern der Malware sein.



Kompetenznachweis 7 vom 11.12.25

In diesem Kompetenznachweis wurde die «Session Fixation» Attacke vertieft angeschaut und mit einem Beispiel in PHP praktisch getestet. Es wurde eine rudimentäre Webseite mit Loginfunktion nachgestellt, wobei die Session ID über die URL übertragen wird. Im Browser schaut die URL beispielsweise folgendermassen aus:



Session ID über die URL im Browser

Um die Sicherheit zu erhöhen und einen Angriff zu erschweren, könnten man z.B. folgende relativ einfache Anpassungen vornehmen:

- Nach dem Login sollte immer eine neue Session ID generiert werden. Dadurch kann die Session ID nicht über die «Session Fixation» Attacke bereits vor dem Login (z.B. über eine Nachricht von einem Angreifer) übergeben werden. PHP stellt hierfür von Haus aus bereits die Funktion session_regenerate_id() zur Verfügung.
- Die Session ID über ein Cookie und nicht über die URL übergeben. So ist die Session ID nicht direkt sichtbar und kann über JavaScript nicht einfach ausgelesen werden.

Der folgende Codeblock zeigt diese zwei Sicherheitsmassnahmen in PHP umgesetzt:

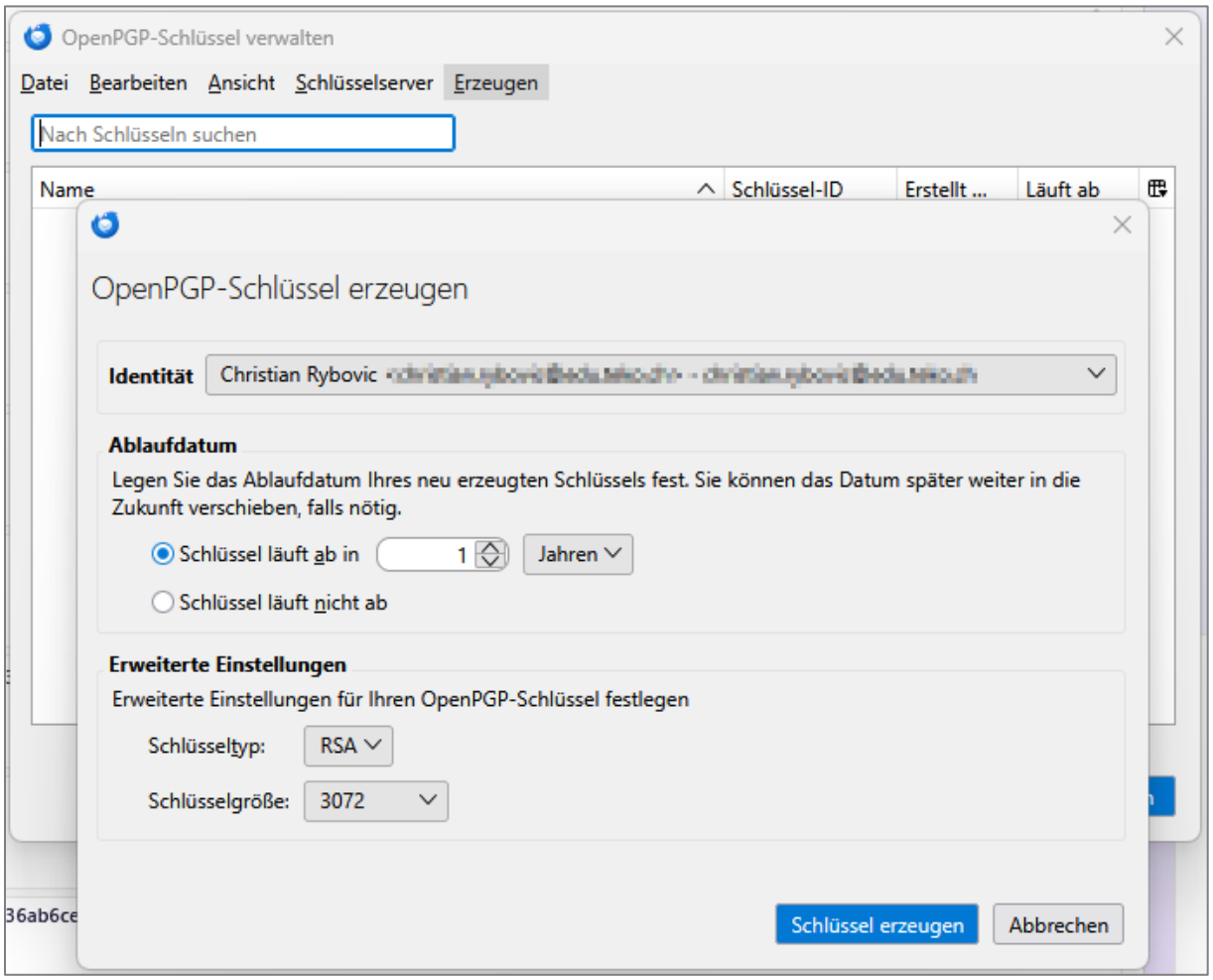
```php
<?php
ini_set('session.use_only_cookies', 1);

session_set_cookie_params([
    'lifetime' => 0,
    'path' => '/',
    'secure' => true,              // Use HTTPS
    'httponly' => true,            // Block JavaScript
    'samesite' => 'Strict'
]);

session_start();

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $username = $_POST['username'] ?? '';
    $password = $_POST['password'] ?? '';
    if ($username === 'admin' && $password === '@jH232-') {
        session_regenerate_id(true);
        $_SESSION['logged_in'] = true;
        $_SESSION['username']  = $username;
        echo 'Login successfull';
    } else {
        echo 'Login failed';
    }
}
```

Kompetenznachweis 8 vom 18.12.25

Die Verschlüsselung von Mails ist gerade bei Unternehmen, Behörden oder im Gesundheitswesen besonders relevant. Ziel dieser Verschlüsselung ist es, die Vertraulichkeit, Integrität und Authentizität der übertragenen Daten sicherzustellen. Hierbei unterscheidet man grundsätzlich zwischen Transportverschlüsselung und Ende-zu-Ende-Verschlüsselung.

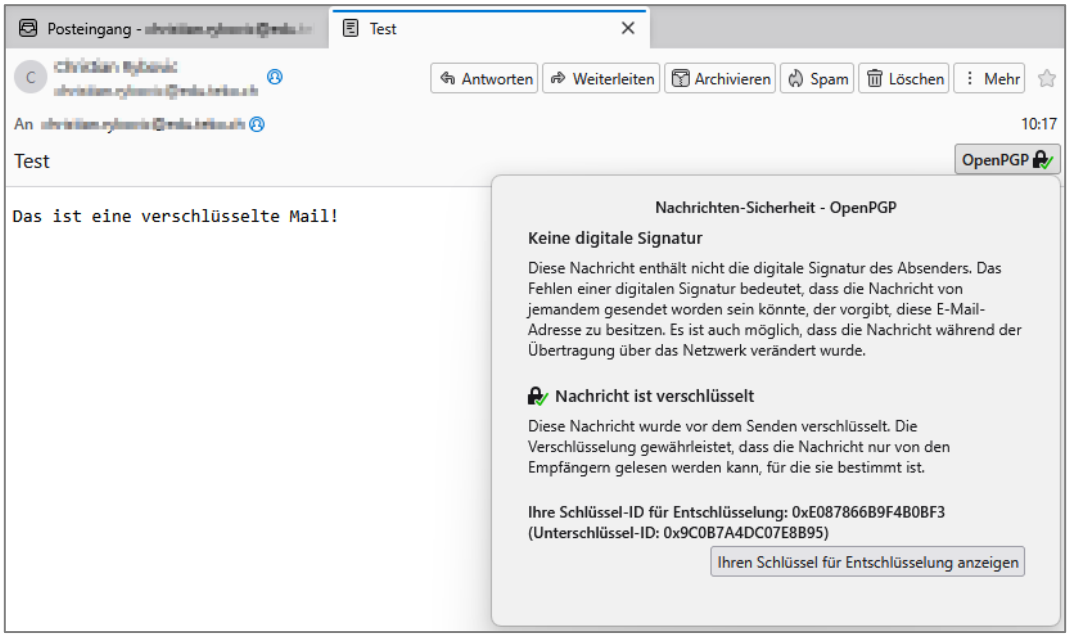
In einer Testumgebung wurde das Verschlüsseln von Mails mit OpenPGP und Thunderbird getestet. Seit Version 78 wird die Verschlüsselungstechnologie OpenPGP von Thunderbird unterstützt und ab Version 78.2.1 ist diese standardmässig aktiviert.

Über die Option «Werkzeuge» und «OpenPGP-Schlüssel verwalten» kann in Thunderbird mit wenigen Klicks ein neues Schlüsselpaar erzeugt werden. Die Standardeinstellungen passen grundsätzlich, in diesem Beispiel wurde das Ablaufdatum für den neu generierten Schlüssel nur auf 1 Jahr verkürzt.



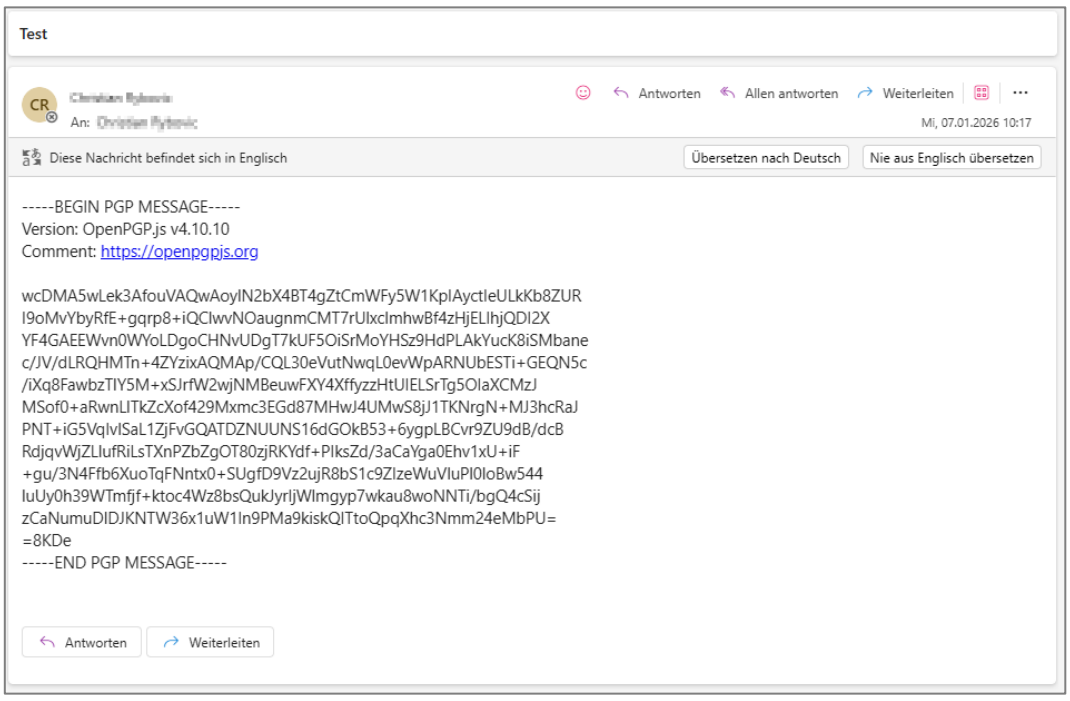
Schlüsselgenerierung in Thunderbird

Der öffentliche Schlüssel kann dann mit den Kontakten geteilt werden. Diese können mithilfe des Schlüssels ein Mail Ende-zu-Ende verschlüsseln. Nachdem das Mail empfangen wird, entschlüsselt Thunderbird automatisch das verschlüsselte Mail.



Mail in Thunderbird

Betrachtet man das Mail z.B. im Browser ohne automatische Entschlüsselung, dann kann man die Nachricht nicht lesen.



Mail im Browser