

Contents

1. Editorial
2. scip AG Informationen
3. Neue Sicherheitslücken
4. Fachartikel
5. Kreuzworträtsel
6. Impressum

1. Editorial

Täglich grüsst die IT-Security

Es ist 06:02 Uhr. Der Wecker klingelt in das dunkle Schlafzimmer. Nach einer kurzen Realisierungsphase betätige ich den Stummschalter.

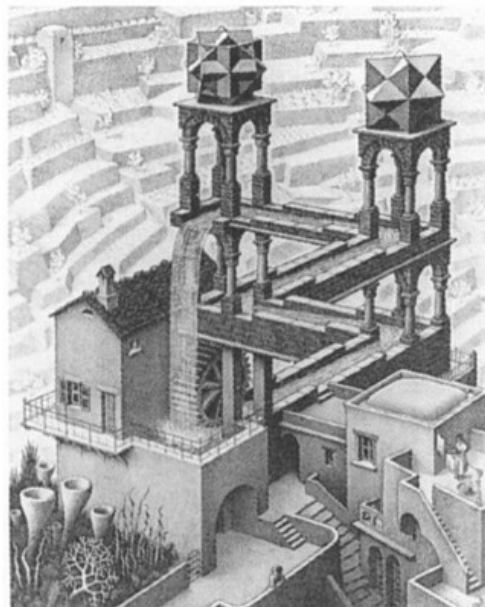
Nach der routinemässigen Morgenpflege – Dusche, Zähneputzen – fahre ich mit meinem Auto in den Technopark Zürich. Das Radio dudelt belanglos vor sich hin und bringt die immer gleichen Musikinterpretationen – U2, Brian Adams und wie sie alle heissen – die Verkehrsfunkdurchsage wiederholt sich mit den gleichen Staumeldungen – Hauptstrasse Bremgarten in Richtung Zürich, Zürich Milchbuck – und der Moderator versucht krampfhaft fröhlich und gut gelaunt zu sein.

Ich parkiere auf dem Firmenparkplatz, besteige den Lift, folge dem Verlauf des Korridores und betrete die scip AG Büroräumlichkeiten im 2.OG des Technoparks. Der Computer wird eingeschaltet und ein extra starker Kaffee tröpfelt, Kapsel sei Dank, mit feinem Schäumchen in meine Tasse.

Mein Arbeitstag beginnt.

In der Arbeitswelt angekommen checke ich meine E-Mails und betrachte den heutigen Terminplan. Auf 10:00 Uhr ist ein Treffen mit einem Kunden eingeplant – Nachbesprechung PenTest Resultate. Vor der Abfahrt in Richtung Kundenmeeting, geniesse ich noch den Kaffee und beantworte ausstehende E-Mails.

Im betroffenen Projekt ging es um die Going-Live Freigabe einer Webapplikation, aus Sicht der IT-Security. Dieses Going-Live musste, im Anschluss an unseren Penetration Test, verschoben werden. Nebst direkten Besprechungen, der Penetration Test Resultate, mit dem Kunden, führen wir, im Bedarfsfall, Nachbesprechungen mit den zuständigen Partnerfirmen durch. Partnerfirmen sind Softwarehersteller, Provider, Integrationshäuser usw. Es ist wichtig, dass Partnerfirmen die vorhandenen Bedrohungen verstehen und die Notwendigkeit der erforderlichen Massnahmen erkennen um diese schnellstmöglich umsetzen. Dem Kunden entsteht damit ein hoher Nutzen.



Der Tisch, im repräsentativen Sitzungsraum des Kunden, ist belegt mit Laptops, Schreibmappen, Kaffeetassen, Wasserflaschen, Gläsern und Visitenkarten. Die Nachbesprechung kann, mit einer kleinen Verspätung, beginnen.

Der Projektleiter des Kunden begrüsst alle Teilnehmer und steckt den Rahmen der kommenden zwei Stunden ab. Ich, als Projektleiter seitens scip AG, erhalte das Wort und die Aufmerksamkeit der Teilnehmer. Die gesammelten sensitiven Daten – anonymisiert – aus den durchgeführten Proof of Concepts werden präsentiert. Das missbrauchte Einfallstor und die angewandte Methodik zur Erlangung der Daten wird schematisch erläutert. Die erforderlichen Sofortmassnahmen werden aufgelistet und in einen Kontext mit den zu planenden

Festigungsmassnahmen gebracht. Das Wort wird der Partnerfirma übergeben. Zeitdruck, Moving Targets, Produkte, Zuständigkeiten etc. werden angemerkt – die Diskussionrunde ist eröffnet. Der federführende Prüfer bespricht die Vorgehensweise mit dem zuständigen Techniker der Partnerfirma. Dieser agiert sehr professionell und anerkennt schnell die gefundenen und ausgenutzten Schwachstellen. Nun werden die Weichen für die weitere Zusammenarbeit, Zuständigkeiten und die Termine zur Fixierung der offenen Punkte koordiniert. Der Projektleiter des Kunden beendet, mit einer kleinen Zusammenfassung der soeben definierten Massnahmen, die Sitzung und die Teilnehmer machen sich auf den Weg in ihre Büros.

Ich parkiere auf dem Firmenparkplatz, besteige den Lift, folge dem Verlauf des Korridores und betrete die scip AG Büroräumlichkeiten im 2.OG des Technoparks...

Simon Zumstein <sizu at scip.ch>
Geschäftsleiter
Zürich, 16. März 2005

2. scip AG Informationen

2.1 Technical Session – Webapplication Security



Ein Eintrittstor in Firmennetze stellen oft Webapplikationen dar. Es gibt eine Vielzahl von Angriffstechniken, die für das System, deren Benutzer und auch die umliegenden Systeme eine grosse Bedrohung ausüben.

Am **Donnerstag dem 07. April 2005** führen wir, in Zusammenarbeit mit NetProtect AG (<http://www.netprotect.ch>), die erste Technical Session mit dem Thema Application Security durch.

Wir haben dazu eine Schulungsumgebung mit verschiedenen Webplattformen (E-Banking Plattformen auf .NET & PHP sowie anderen Technologien & diverse Webportale unter IIS & Apache) aufgebaut, in denen die vielfältigen Angriffsarten simuliert werden können.

Dabei werden unter anderem diese Problembereiche ausgeführt:

- Input-Validierung: Cross-Site Scripting & OS
- Command Injection
- Session State Management
- Session Management: Session Hijacking
- Parameter Manipulation: Hidden Field Tampering
- etc.

Details zum Workshop und Anmeldemöglichkeiten finden Sie über unsere Webpage <http://www.scip.ch> im Untermenü Workshops oder via folgendem Link:
http://www.scip.ch/dienstleistungen/workshops/vertiefung/ts_webapplication/

Machen Sie mit beim Kreuzworträtsel auf der Seite 12 und gewinnen Sie einen Platz im **Technical Session – Webapplication Security!**

3. Neue Sicherheitslücken

Die erweiterte Auflistung hier besprochener Schwachstellen sowie weitere Sicherheitslücken sind unentgeltlich in unserer Datenbank unter <http://www.scip.ch/cgi-bin/smss/showadvf.pl> einsehbar.

Die Dienstleistungspakete [\)scip\(pallas](#) liefern Ihnen jene Informationen, die genau für Ihre Systeme relevant sind.

Contents:

- 3.1 Microsoft Windows XP und Server 2003 Land Denial of Service
- 3.2 X11 bis 6.x libXpm XPM-Bild Pufferüberlauf
- 3.3 TYPO3 CMW Linklist Extension bis 1.4.2 category_uid SQL Injection
- 3.4 Real Networks RealPlayer bis 10.x und RealOne Player bis v2 SMIL-Dateien Pufferüberlauf
- 3.5 Real Networks RealPlayer bis 10.x und RealOne Player bis v2 WAV-Dateien Pufferüberlauf
- 3.6 Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 SSL Anzeige vortäuschen
- 3.7 Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 UTF8 zu Unicode Konvertierung Pufferüberlauf
- 3.8 Mozilla bis 1.7.6, Mozilla Firefox bis 1.0.1 und Thunderbird 1.0.1 nsTSubstring_CharT::Replace() Pufferüberlauf
- 3.9 Mozilla Firefox bis 1.0.1 Form AutoComplete erweiterte Leserechte
- 3.10 Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 anderer Tab htaccess-Authentisierung vortäuschen
- 3.11 PHP bis 4.3.x readfile() Denial of Service
- 3.12 TrendMicro Antivirus-Produkte ARJ-Archive Pufferüberlauf
- 3.13 phpMyAdmin bis 2.6.1-pl1 phpmyadmin.css.php und database_interface.lib.php erweiterte Leserechte
- 3.14 Microsoft Internet Explorer bis 6.0 Pop-up-Fenster URL-Anzeige vortäuschen Mozilla Firefox bis 1.0 International Domain Name Seiten-Informationen vortäuschen

3.1 Microsoft Windows XP und Server 2003 Land Denial of Service

Einstufung: **kritisch**
 Remote: Ja
 Datum: 08.03.2005
 scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1268>

Microsoft Windows XP stellt eine Weiterentwicklung des professionellen Windows 2000 dar. Microsoft Windows 2003 Server ist die Server-Version von Windows XP. In den neuen Windows-Versionen wurde eine klassische Denial of Service-Möglichkeit namens Land bestätigt. Diese Angriffsart ist seit Ende 1997 bekannt, und es handelt sich hierbei um eine der letzten Varianten der DoS-Attacken, welche beachtliche Popularität erlangte. Durch die Land-Attacke wird ein sehr komplexer Angriff ausgeführt, welcher ein SYN-Paket mit identischer Absender- und Empfängerport erzeugt. Anschliessend wird dieses Paket an einen offenen Port gesendet, wo das Paket durch die vielen IP-Stacks eine Art Race-Condition erzeugt, und dadurch das System des Opfers lahmlegt. Durch Firewalling sollten entsprechende Pakete gefiltert werden, bis Microsoft mit einem Patch reagiert.

Expertenmeinung:

Wirklich verwunderlich, dass eine solch alte Schwachstelle nach Jahren in einer populären Betriebssystem-Reihe wieder auftaucht. Dies lässt ernsthafte Zweifel an der Kompetenz der Microsoft-Entwickler aufkommen, die scheinbar einmal mehr den gleichen Fehler machten. Es ist nun also mit einer Vielzahl an neuen Land-Angriffen zu rechnen. Vor allem Systeme, die direkt über das Internet erreichbar sind, könnten Opfer entsprechender Skript-Kiddie Attacken werden. Firewalling tut deshalb dringend not.

3.2 X11 bis 6.x libXpm XPM-Bild Pufferüberlauf

Einstufung: **kritisch**
 Remote: Indirekt
 Datum: 04.03.2005
 scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1266>

X11 stellt eine Schnittstelle zwischen der Hardware (z.B. Tastatur, Maus und Bildschirm) und der virtuellen Arbeitsumgebung dar. Zu diesem Zweck wird eine einheitliche API zur Verfügung gestellt. X11 wurde von allen großen Computer-Herstellern als Grundlage eines eigenen Fenstersystems übernommen und so

quasi zu einem Standard erklärt. Der Linux Distributor Gentoo weist in seinem GLSA 200503-08 auf eine Pufferüberlauf-Schwachstelle von X11 bis 6.x libXpm bei der Verarbeitung von XPM-Bildern hin. Im Gentoo-Advisory wird in erster Linie die Denial of Service-Möglichkeit hervorgehoben. Ebenfalls sei aber das Ausführen beliebigen Programmcodes mit erweiterten Rechten denkbar. Ein Workaround existiert nicht - Das Problem wurde jedoch in einer aktuellen Version von OpenMotif behoben.

Expertenmeinung:

Diese Schwachstelle ist relativ kritisch, da die Verbreitung von XFree86 durch die vielen Linux-Distributionen sehr hoch ist. Die Zeit wird jedoch erst zeigen, wie sich die Verbreitung der erfolgreichen Angriffe entwickeln wird. Man sollte jedoch auf Nummer Sicher gehen und die aktualisierte Version von X11 bzw. OpenMotif installieren.

3.3 TYPO3 CMW Linklist Extension bis 1.4.2 category_uid SQL Injection

Einstufung: **kritisch**
Remote: Ja
Datum: 04.03.2005
scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1265>

TYPO3 ist ein populäres open-source Content Management System (CMS), das vorwiegend im professionellen Bereich eingesetzt wird. Es gibt eine Vielzahl an Extensions, mit denen die Core-Funktionalität von TYPO3 erweitert werden kann. Die Linklist Extension bis 1.4.2 ist gegen eine SQL Injection verwundbar. Dies ist über den Parameter category_uid möglich, der zur Auswahl der Link-Kategorie genutzt wird. Ein Angreifer sieht sich über den Fehler in der Lage, erweiterte Rechte auf der Datenbank-Ebene zu erhalten. Der Fehler wurde in der jüngsten Version von TYPO3 CMW Linklist Extension behoben.

Expertenmeinung:

Schwerwiegende Schwachstellen in Content Management Systemen (CMS) sind bei Angreifern gern gesehen. Sehr schnell und unkompliziert lassen sich so eine Vielzahl an Web-Systemen kompromittieren. In den weniger schlimmen Fällen wird lediglich ein Defacement umgesetzt. Weit aus tragischer wird es, wenn Hintertüren eingerichtet und im Geheimen sensitive Daten kopiert werden. Betroffene Webmaster sollten deshalb unverzüglich Gegenmassnahmen umsetzen.

3.4 Real Networks RealPlayer bis 10.x und RealOne Player bis v2 SMIL-Dateien Pufferüberlauf

Einstufung: **kritisch**
Remote: Ja
Datum: 01.03.2005
scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1263>

Der Real Player der Firma Real Networks kann für das Abspielen der hauseigenen Real-Formate (RealAudio und RealVideo) genutzt werden. Die Software ist als Freeware-Version für Windows, UNIX und Macintosh verfügbar. Gleich zwei Pufferüberlauf-Schwachstellen bei der Verarbeitung von Multimedia-Dateien durch die RealPlayer-Reihe von Real Networks wurden gefunden. Eine davon betrifft die SMIL-Dateien (Synchronized Multimedia Integration Language). Wenige technische Details sind bekannt - Ein Exploit zur Schwachstelle wurde jedoch nicht publiziert. Real Networks hat für sämtliche betroffenen Produkte einen Patch herausgegeben, der vorzugsweise über das AutoUpdate-Feature bezogen werden kann.

Expertenmeinung:

Ähnlich der Pufferüberlauf-Schwachstellen in anderen Playern zeigt auch diese Verwundbarkeit, dass harmlose Client-Applikationen für schwerwiegende Attacken missbraucht werden können. Das Einspielen der Patches bzw. das Updaten auf die aktuellste Player-Version ist entsprechend empfohlen.

3.5 Real Networks RealPlayer bis 10.x und RealOne Player bis v2 WAV-Dateien Pufferüberlauf

Einstufung: **kritisch**
Remote: Ja
Datum: 01.03.2005
scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1262>

Der Real Player der Firma Real Networks kann für das Abspielen der hauseigenen Real-Formate (RealAudio und RealVideo) genutzt werden. Die Software ist als Freeware-Version für Windows, UNIX und Macintosh verfügbar. Gleich zwei Pufferüberlauf-Schwachstellen bei der Verarbeitung von Multimedia-Dateien durch die RealPlayer-Reihe von Real Networks wurden gefunden. Eine davon betrifft das klassische WAV-Format zum Abspielen von Sounddateien. Wenige technische Details sind bekannt - Ein Exploit zur Schwachstelle wurde jedoch nicht publiziert. Real Networks hat für sämtliche

betroffenen Produkte einen Patch herausgegeben, der vorzugsweise über das AutoUpdate-Feature bezogen werden kann.

Expertenmeinung:

Ähnlich der Pufferüberlauf-Schwachstellen in anderen Playern zeigt auch diese Verwundbarkeit, dass harmlose Client-Applikationen für schwerwiegende Attacken missbraucht werden können. Das Einspielen der Patches bzw. das Updaten auf die aktuellste Player-Version ist entsprechend empfohlen.

3.6 Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 SSL Anzeige vortäuschen

Einstufung: **kritisch**

Remote: Ja

Datum: 01.03.2005

scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1259>

Das Mozilla-Projekt versucht einen open-source Webbrowser zur Verfügung zu stellen. Der Mozilla Webbrowser erlangte seit der Veröffentlichung der ersten offiziellen Versionen immer mehr Akzeptanz, kann jedoch in Punkto Marktanteile mit den anderen Grössen des Genres noch nicht mithalten. Im Mozilla Foundation Security Advisory 2005-14 gibt der Hersteller eine Schwachstelle in Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 bekannt. Es gibt verschiedene Möglichkeiten, wie das SSL-Symbol angezeigt werden kann, obwohl keine sichere Verbindung besteht. Eine Möglichkeit besteht darin, nach einer sicheren Seite eine unsichere zu öffnen, die jedoch nie fertiggeladen wird. Das SSL-Lock wird sodann nicht aktualisiert. Eine andere Möglichkeit besteht durch das Generieren eines sicheren Fensters mittels `document.write()` oder die Server-Rückgabe HTTP 204 zu falschen Darstellung. Ein Exploit zur Schwachstelle ist nicht bekannt. Das Mozilla Team empfiehlt das Upgrade auf eine aktualisierte Software-Version. Als weitere Lösung ist zur Zeit der Wechsel zu einem alternativen Produkt (z.B. Opera) anzusehen.

Expertenmeinung:

Eine Vielzahl an Sicherheitslücken prasselt auf das Mozilla-Projekt nieder. Keine gute Werbung, in der Tat - Obschon Mozilla immerwieder als Alternative zum beschmutzten Microsoft Internet Explorer empfohlen wird, wird voraussichtlich über längere Zeit auch das Mozilla-Pendant seine weisse Weste nicht sauber halten können. Es scheint, als müsse der sichere Webbrowser erst noch entwickelt werden, denn Mozilla bringt die gleichen (Kinder-)Krankheiten mit, wie auch schon viele vergleichbare Projekte zuvor.

3.7 Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 UTF8 zu Unicode Konvertierung Pufferüberlauf

Einstufung: **kritisch**

Remote: Ja

Datum: 01.03.2005

scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1258>

Das Mozilla-Projekt versucht einen open-source Webbrowser zur Verfügung zu stellen. Der Mozilla Webbrowser erlangte seit der Veröffentlichung der ersten offiziellen Versionen immer mehr Akzeptanz, kann jedoch in Punkto Marktanteile mit den anderen Grössen des Genres noch nicht mithalten. Im Mozilla Foundation Security Advisory 2005-15 gibt der Hersteller eine Schwachstelle in Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 bekannt. So besteht ein Heap Overflow beim Konvertieren von UTF8 zu Unicode. Dieser erlaubt das Ausführen beliebigen Programmcodes. Genaue technische Details oder ein Exploit zur Schwachstelle sind nicht bekannt. Das Mozilla Team empfiehlt das Upgrade auf eine aktualisierte Software-Version. Als weitere Lösung ist zur Zeit der Wechsel zu einem alternativen Produkt (z.B. Opera) anzusehen.

Expertenmeinung:

Eine Vielzahl an Sicherheitslücken prasselt auf das Mozilla-Projekt nieder. Keine gute Werbung, in der Tat - Obschon Mozilla immerwieder als Alternative zum beschmutzten Microsoft Internet Explorer empfohlen wird, wird voraussichtlich über längere Zeit auch das Mozilla-Pendant seine weisse Weste nicht sauber halten können. Es scheint, als müsse der sichere Webbrowser erst noch entwickelt werden, denn Mozilla bringt die gleichen (Kinder-)Krankheiten mit, wie auch schon viele vergleichbare Projekte zuvor.

3.8 Mozilla bis 1.7.6, Mozilla Firefox bis 1.0.1 und Thunderbird 1.0.1 nsTSubstring_CharT::Replace() Pufferüberlauf

Einstufung: **kritisch**

Remote: Ja

Datum: 01.03.2005

scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1256>

Das Mozilla-Projekt versucht einen open-source Webbrowser zur Verfügung zu stellen. Der Mozilla Webbrowser erlangte seit der Veröffentlichung der ersten offiziellen Versionen immer mehr Akzeptanz, kann jedoch in Punkto

Marktanteile mit den anderen Grössen des Genres noch nicht mithalten. Im Mozilla Foundation Security Advisory 2005-18 gibt der Hersteller eine Schwachstelle in Mozilla bis 1.7.6, Mozilla Firefox bis 1.0.1 und Thunderbird 1.0.1 bekannt. Durch einen Pufferüberlauf-Fehler bei String liesse sich unter Umständen beliebiger Programmcode ausführen. Genaue technische Details oder ein Exploit zur Schwachstelle sind nicht bekannt. Das Mozilla Team empfiehlt das Upgrade auf eine aktualisierte Software-Version. Als weitere Lösung ist zur Zeit der Wechsel zu einem alternativen Produkt (z.B. Opera) anzusehen.

Expertenmeinung:

Eine Vielzahl an Sicherheitslücken prasselt auf das Mozilla-Projekt nieder. Keine gute Werbung, in der Tat - Obschon Mozilla immerwieder als Alternative zum beschmutzten Microsoft Internet Explorer empfohlen wird, wird voraussichtlich über längere Zeit auch das Mozilla-Pendant seine weisse Weste nicht sauber halten können. Es scheint, als müsse der sichere Webbrowser erst noch entwickelt werden, denn Mozilla bringt die gleichen (Kinder-)Krankheiten mit, wie auch schon viele vergleichbare Projekte zuvor.

3.9 Mozilla Firefox bis 1.0.1 Form AutoComplete erweiterte Leserechte

Einstufung: **kritisch**

Remote: Ja

Datum: 01.03.2005

scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1254>

Das Mozilla-Projekt versucht einen open-source Webbrowser zur Verfügung zu stellen. Der Mozilla Webbrowser erlangte seit der Veröffentlichung der ersten offiziellen Versionen immer mehr Akzeptanz, kann jedoch in Punkto Marktanteile mit den anderen Grössen des Genres noch nicht mithalten. Im Mozilla Foundation Security Advisory 2005-19 gibt der Hersteller eine Schwachstelle in Mozilla Firefox bis 1.0.1 bekannt. Das AutoComplete-Feature für Web-Forms weist ein Problem auf. Und zwar kann eine Webseite die vorgeschlagenen Werte auslesen, noch bevor die Daten durch den Benutzer wirklich freigegeben wurden. Genaue technische Details oder ein Exploit zur Schwachstelle sind nicht bekannt. Das Mozilla Team empfiehlt das Upgrade auf eine aktualisierte Software-Version. Als weitere Lösung ist zur Zeit der Wechsel zu einem alternativen Produkt (z.B. Opera) anzusehen.

Expertenmeinung:

Eine Vielzahl an Sicherheitslücken prasselt auf das Mozilla-Projekt nieder. Keine gute Werbung, in der Tat - Obschon Mozilla immerwieder als Alternative zum beschmutzten Microsoft Internet Explorer empfohlen wird, wird voraussichtlich über längere Zeit auch das Mozilla-Pendant seine weisse Weste nicht sauber halten können. Es scheint, als müsse der sichere Webbrowser erst noch entwickelt werden, denn Mozilla bringt die gleichen (Kinder-)Krankheiten mit, wie auch schon viele vergleichbare Projekte zuvor.

3.10 Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 anderer Tab htaccess-Authentisierung vortäuschen

Einstufung: **kritisch**

Remote: Ja

Datum: 01.03.2005

scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1251>

Das Mozilla-Projekt versucht einen open-source Webbrowser zur Verfügung zu stellen. Der Mozilla Webbrowser erlangte seit der Veröffentlichung der ersten offiziellen Versionen immer mehr Akzeptanz, kann jedoch in Punkto Marktanteile mit den anderen Grössen des Genres noch nicht mithalten. Im Mozilla Foundation Security Advisory 2005-24 gibt der Hersteller eine Schwachstelle in Mozilla bis 1.7.6 und Mozilla Firefox bis 1.0.1 bekannt. Für eine Webseite ist es möglich, eine htaccess-Authentisierung auf einem anderen Tab erscheinen zu lassen. Ein Benutzer kann so zur Eingabe sensibler Informationen, die dadurch mitgelesen und später weiterverwendet werden können, bewegt werden. Genaue technische Details oder ein Exploit zur Schwachstelle sind nicht bekannt. Das Mozilla Team hat keinen Patch angekündigt, jedoch einen Workaround bekannt gegeben: Es sollen keine Webseiten zwielichtigen Inhalts angesteuert sowie nur Passwort-Eingaben auf vertrauenswürdigen und zweifelsfrei feststellbaren Angeboten umgesetzt werden. Als weitere Lösung ist zur Zeit der Wechsel zu einem alternativen Produkt (z.B. Opera) anzusehen.

Expertenmeinung:

Eine Vielzahl an Sicherheitslücken prasselt auf das Mozilla-Projekt nieder. Keine gute Werbung, in der Tat - Obschon Mozilla immerwieder als Alternative zum beschmutzten Microsoft Internet Explorer empfohlen wird, wird voraussichtlich über längere Zeit auch das Mozilla-Pendant seine weisse Weste nicht sauber halten können. Es scheint, als müsse der sichere Webbrowser

erst noch entwickelt werden, denn Mozilla bringt die gleichen (Kinder-)Krankheiten mit, wie auch schon viele vergleichbare Projekte zuvor.

3.11 PHP bis 4.3.x readfile() Denial of Service

Einstufung: **kritisch**
Remote: Ja
Datum: 25.02.2005
scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1246>

PHP ist ein frei verfügbares open-source Skripting-Paket, das für sämtliche populären Betriebssysteme zur Verfügung steht. Es findet vorwiegend im Web-Einsatz seine Verwendung. SuSE berichtet in SUSE-SR:2005:006 von einer Denial of Service-Schwachstelle in PHP bis 4.3.x. Die Funktion readfile() kann zu einem Absturz des PHP-Systems führen. Dies geschieht dann, wenn über die besagte Funktion eine Datei mit einer Grösse des Vielfachen der Page-Size des genutzten Betriebssystems angesteuert wird. Genaue technische Details oder ein Exploit zur Schwachstelle sind nicht bekannt. Als Workaround wird empfohlen, Zugriffe auf PHP-Dokumente mit readfile() zu limitieren.

Expertenmeinung:

Eine sehr interessante Sicherheitslücke, die auf den ersten Blick sehr hypothetisch wirkt. Es ist jedoch tatsächlich möglich, diesen Angriff auf betroffenen Systemen mit einem gewissen Aufwand umzusetzen. Umso wichtiger ist es, nun seine Systeme bei Gelegenheit gegen diesen Angriff zu schützen.

3.12 TrendMicro Antivirus-Produkte ARJ-Archive Pufferüberlauf

Einstufung: **kritisch**
Remote: Ja
Datum: 25.02.2005
scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1243>

TrendMicro stellt eine Antiviren-Lösung für verschiedene Plattformen und variablen Varianten zur Verfügung. Wie ISS ursprünglich meldete, existiert in den verschiedenen Antiviren-Produkten von Trend Micro eine Pufferüberlauf-Schwachstelle bei der Verarbeitung von ARJ-Archiven. Ein Angreifer kann über eine korrupte ARJ-Datei Programmcode auf einem betroffenen System ausführen. Technische Details oder ein Exploit zur Schwachstelle sind nicht bekannt. Trend Micro hat die aktualisierte Scan Engine VSAPI 7.510 für die betroffenen Produkte zum

Download bereitgestellt.

Expertenmeinung:

Dieses Problem hat auch schon die Symantec-Lösungen betroffen. Das Ganze entwickelt sich also langsam aber sicher zu einer wirklich schwerwiegende da flächendeckende Schwachstelle. Die Entwickler von Computerviren kommen deshalb in den Genuss, neue Systeme erschliessbar machen zu können. Gegenmassnahmen sind zur Prävention von grossangelegten Viren-Übergriffen von erhöhter Dringlichkeit.

3.13 phpMyAdmin bis 2.6.1-pl1 phpmyadmin.css.php und database_interface.lib.php erweiterte Leserechte

Einstufung: **kritisch**
Remote: Ja
Datum: 22.02.2005
scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1242>

phpMyAdmin ist eine beliebte Web-Oberfläche für die PHP-Administration. Maksymilian Arciemowicz entdeckte zwei Schwachstellen in phpMyAdmin bis 2.6.1-pl1. Eine davon betrifft die PHP-Dokumenten phpmyadmin.css.php und database_interface.lib.php. Eingaben werden nicht auf ihre Gültigkeit hin überprüft, so dass erweiterte Leserechte für lokale Dateien umgesetzt werden kann. Ein erfolgreicher Angriff setzt aktiviertes register_globals und deaktiviertes magic_quotes_gpc voraus. Genaue technische Details zur Schwachstelle oder ein Exploit sind nicht bekannt. Der Fehler wurde in phpMyAdmin 2.6.1-pl1 behoben.

Expertenmeinung:

phpMyAdmin läuft meist in einem geschützten Bereich ab und ist nur legitimen Administratoren zugänglich. Das Umsetzen von erfolgreichen Cross Site Scripting-Attacken durch einen externen Angreifer gestalten sich deshalb nicht gerade einfach. Das Interesse an dieser Schwachstelle ist deshalb eher akademischer Natur.

3.14 Microsoft Internet Explorer bis 6.0 Popup-Fenster URL-Anzeige vortäuschen

Einstufung: **kritisch**
Remote: Ja
Datum: 21.02.2005
scip DB: <http://www.scip.ch/cgi-bin/smss/showadvf.pl?id=1239>

Der Microsoft Internet Explorer (MS IEX) ist der am meisten verbreitete Webbrowser und fester Bestandteil moderner Windows-Betriebssysteme. Standardmässig forciert seit dem Service Pack 2 von Microsoft Windows XP der Internet Explorer die URL in der Titelzeile eines Popup-Fensters ohne Adresszeile. Dadurch sollen Phishing-Angriffe limitiert werden. bitlance winter entdeckte nun, dass diese URL-Angabe durch das Nutzen eines überlangen Hostnamens in der URL umgangen werden kann. Bisher sind keine Gegenmassnahmen bekannt. Es wird empfohlen lediglich Links aus bekannter Herkunft zu folgen und auf die Angabe sensibler Daten in Popup-Fenstern (ohne Adresszeile) gänzlich zu verzichten.

Expertenmeinung:

Es ist nur eine Frage der Zeit, bis diese Schwachstelle breitflächig in den Markt der Spam-Mails Einzug halten wird. Ahnungslose Benutzer können so in bester Phishing-Manier zum Besuch eines zwielichtigen Angebots bewegt werden. Das Mitschneiden von Passwort-Eingaben oder das Umsetzen von manipulierten Transaktionen ist sodann nur noch ein Kinderspiel. Microsoft täte gut daran, sich unverzüglich der Sicherheit ihres Webbrowsers zu widmen. Der Image-Schaden wird - egal wie gut Windows Longhorn ausfallen wird - nicht mehr so schnell wettzumachen sein.

4. Fachartikel

4.1 Warum sich der Bereich der Computersicherheit zurückentwickelt

Marc Ruef

"Sie sind schuld!" So oder so ähnlich klang der Vorwurf, den mir ein Hersteller einer Sicherheitslösung in einem Telefonat an den Kopf warf.



Aber lassen Sie mich die Geschichte von vorne beginnen. An einem kühlen Morgen des letzten Herbsts fand ich mich im Server-Raum einer unserer Kunden wieder. Eine Privatbank aus Zürich, die ein sehr gepflegtes Netzwerk ihr eigen nennt. Während mir die Server die heisse Luft ihres Innenlebens entgegenbliesen, hackte ich wie wild auf meiner Tastatur herum. Meine Aufgabe in diesem schon fast menschenfeindlichen Umfeld, bestehend aus riesen Blechkisten war es, eine potentielle Sicherheitslücke in der frisch eingeführten E-Banking Lösung zu finden. Na ja, nicht das erste Mal, dass man mich auf so etwas loslässt und meine Erfahrung hat entsprechend meinen Optimismus nur gestärkt, dass ich wohl oder übel etwas finden werde.

Die Stunden vergingen, als ich denn so vor meinen VMware-Geräten sass und mich durch unendliche Zeilen an Programmcode und Ausgaben meiner Analyse-Tools wühlte. Plötzlich fiel mir etwas auf: Die Authentisierung des Benutzers findet in einem ersten Schritt mittels Benutzername und Passwort über die Web-Schnittstelle statt. Es war mir nun möglich zu bestimmen, wohin der Benutzer nach dem Durchlaufen dieser Authentisierungs-Prozedur landen würde. Das Umsetzen von Phishing-Attacken war nun ein Kinderspiel, denn ich konnte jeden Online-Kunden der Bank auf meine eigenen Server locken und ihm dort die sensitiven Kunden-Daten aus der Tasche ziehen. Heureka, Auftrag erfolgreich erledigt!

Nach dem eindeutigen Verifizieren meiner Vermutung besprach ich das Problem kurz mit dem Security Officer der Bank. Ich schilderte ihm die Möglichkeiten und taxierte die Schwachstelle als ernstzunehmendes Risiko für die Reputation der gesamten Bank. Er stimmte mir zu. Ich schlug vor, dass ich mich sofort mit dem Hersteller der betroffenen Lösung - namentlich

Netegrity - in Verbindung setzen würde, damit so schnell wie möglich Gegenmassnahmen ergriffen werden konnten. Ein Patch oder ein Workaround in der Konfiguration wären ideal, um dem Problem mit möglichst wenig Aufwand von unserer Seite Rechnung zu tragen. Als ich zurück im Büro war, schrieb ich sofort ein Email an Netegrity, in dem ich das Problem kurz schilderte und um eine baldige Rückantwort bat.

Es vergingen keine zwei Stunden, da klingelte mein Telefon. Die Nummern-Anzeige liess ein Anruf aus Frankreich vermuten. Mit gebrochenem Englisch begrüsst mich ein Herr von Netegrity. Ich war froh, dass so schnell jemand reagierte, denn ich lasse meine Kunden nur ungern exponiert im Internet alleine. Mein Gegenüber bedankte sich, dass ich das Problem gemeldet habe. Er wolle nun jedoch wissen, um welchen Kunden es sich handelt. Ich verweigerte eine Auskunft, denn durch das stets mit unseren Kunden abgeschlossene Non Disclosure Agreement (NDA) und das uns auferlegte Schweizer Bankengesetz ist es untersagt, mit Drittpersonen in irgendeiner Weise über unsere Kunden und die Beziehungen mit ihnen zu sprechen. "Wir halten uns daran und eine Hilfestellung muss auch ohne die Weitergabe dieser Information erfolgen", erwiderte ich auf die Frage. Der Ton meines neuen Freundes aus Frankreich wurde schärfer. Er griff mich an und fragte, warum ich mich denn überhaupt mit ihnen in Verbindung gesetzt hätte, wenn ich mich sowieso nicht kooperativ verhalten möchte. Ich wies nocheinmal sachlich darauf hin, dass ich an Abmachungen gebunden sei und zuerst unseren Kunden fragen müsse, ob er eine Preisgabe seines Namens zustimmt. Mein Gesprächspartner meinte darauf hin, dass ich die Bewilligung einholen müsse und er mir ansonsten nicht weiterhelfen würde. Ich bestätigte und legte den Hörer wieder zurück in die Gabel.



Unverzüglich rief ich unseren Kunden an und schilderte ihm den Ablauf des etwas sonderbar

anmutenden Gesprächs. Wie auch ich sah er den Grund nicht, warum Netegrity den Namen des Kunden zur Bearbeitung eines technischen Grundproblems ihrer Lösung wissen wolle. Der Kunde verweigerte die Herausgabe seines Namens und übertrug mir nocheinmal die volle Kompetenz bei der Abarbeitung dieses Falls.

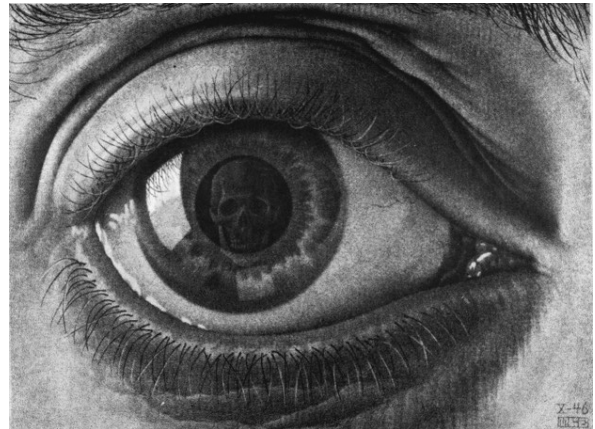
Wiederum rief ich meinen Kontakt bei Netegrity an und teilte ihm mit, dass die Nennung des Kunden nicht möglich sei. Die Abarbeitung des Problems sei sowieso kundenunabhängig und betreffe jegliche Installation von Netegrity SiteMinder. Ich hätte das Problem ja auch bei einem privaten Test per Zufall finden können und in einem solchen Fall könne ich ebenso nicht mit einem Kundennamen aufwarten. Mein zunehmend unfreundlicher werdendes Gegenüber sagte mir schroff, dass er sich nicht mehr mit mir unterhalten wolle. Der Kunde müsse nun über die offizielle Webseite einen Support-Case eröffnen und dort kommt er halt um die Nennung seines Namens bzw. seiner Kundennummer nicht herum.

Einmal mehr rief ich unseren Kunden an und schilderte ihm das letzte Gespräch. Auch er wurde langsam über das Verhalten von Netegrity ungehalten. Er weigerte sich, einen offiziellen Support-Case zu eröffnen, denn die technischen Daten zum Problem habe ich schon bei meinem ersten Email an Netegrity geliefert. Es lag eigentlich nur noch am Hersteller, mir einen Hinweis auf das weitere Vorgehen und eine Terminierung für eine Lösung mitzuteilen. Ich einigte mich mit dem Kunden darauf, dass wir nun noch einige Wochen ins Land gehen lassen würden. Danach setzen wir ein Advisory zum Problem auf, das wir über die jeweiligen Sicherheits-Mailingliste und -Newsseiten verteilen. Durch das Publizieren der Schwachstelle machen wir dem Hersteller Druck, so dass er sich des Problems zwingend annehmen muss.

So war es denn nun auch. Ich setzte ein kleines Advisory auf, das ich an die üblichen Dienste wie Bugtraq, Full-Disclosure, SecuriTeam.com und Secunia schickte. Ebenfalls nahm ich das Problem in unsere hauseigene Verwundbarkeitsdatenbank auf. Die Resonanz auf unser Advisory war interessant. Eine Vielzahl an Firmen, die die gleiche Lösung einsetzen, schrieben uns an und baten um Unterstützung. Sie erbeteten zusätzliche Details zum Problem und fragten, ob wir denn schon einen Workaround ausgearbeitet hätten. Ich merkte, dass die Sache durchaus für so manche Bank ein reales Risiko darstellte.

Ich bin der Meinung, dass ein Hersteller - egal welcher Couleur - das Recht auf eine faire Behandlung bei neu entdeckten Sicherheitsproblemen hat. Dies bedeutet, dass man nach einem Fund sofort den Hersteller informiert und mit ihm das weitere Vorgehen koordiniert. In den meisten Fällen wird er die kommenden Tagen oder Wochen für die Entwicklung eines Patches investieren. Sobald dieser erscheint, wird ein Advisory herausgegeben, der die Benutzer auf das Problem und die möglichen Gegenmassnahmen aufmerksam macht. So kann garantiert werden, dass das Zeitfenster für erfolgreiche Angriffe möglichst klein gehalten wird. Der Hersteller muss nicht erst nach der Veröffentlichung eines Exploits unter Zeitdruck mit einer Gegenmassnahme reagieren.

Das Verhalten von Netegrity hat aber eindeutig bewiesen, dass weder der Kunde noch die Qualität des Produkts von Wichtigkeit ist. Netegrity war einzig und alleine darum bemüht, den Kunden zu identifizieren und ihm die Sache schön zu reden. Dadurch sollten wir als kritische Auditoren umgangen und hinter unserem Rücken Lobbyismus betrieben werden. Es kostet weniger Geld, einen Kunden mit Geschwafel einzulullen, weder einen funktionierenden Patch auszuarbeiten. Der Kapitalismus treibt manchmal sonderbare Früchte.



Dies ist nicht die einzige Geschichte mit einem solchen Ablauf, der aufgrund fehlender Kooperation durch Hersteller mit einem absoluten Nachteil für alle Beteiligten endet. Mir fällt es schon länger auf, dass Sicherheitsprobleme zunehmend durch „mafia-ähnliche“ Methoden bearbeitet werden. Hersteller ignorieren Meldungen zu Schwachstellen, Advisories werden einfach zurückgehalten oder durch einen Marketing-Vertreter schöngeredet, Auditoren werden mit Drohung von Anwälten eingeschüchtert oder in der Öffentlichkeit diffamiert.

Die Konsequenz davon ist verheehrend. Eine Vielzahl an Leuten, die sich in den vergangenen Jahren durch ausgezeichnete Publikationen und clevere Exploits einen Namen gemacht haben, ziehen sich zunehmends aus der Öffentlichkeit zurück. Ihnen fehlt der Wille und die Energie, auch weiterhin auf Schwachstellen hinzuweisen. Der Lohn, der ihnen gezollt wird, ist ja sowieso nur Ignoranz oder gar Aggressivität. Die Anzahl der technisch fundierten Advisories hat in letzter Zeit abgenommen - Ich zweifle aber daran, dass auch wirklich weniger Sicherheitslücken in den eingesetzten Produkten gegeben sind. Die Einschüchterung der Hersteller scheint zu funktionieren. Die "freien Journalisten der Computersicherheit" sind zum Grossteil mundtot gemacht. Zustände, die an die staatliche Zensur in China erinnern.

Das Ganze macht mich traurig, denn der Kapitalismus hat sich eine Welt geschaffen, in der er sich selber legitimiert. Dadurch schützt er seine eigenen Interessen und kann fast nicht mehr durchbrochen werden. Vor allem dann, wenn das EU-Parlament irgendwelche dubiosen Gesetze verabschiedet, die es verbieten, sich mit Sicherheitslücken in Computersystemen auseinanderzusetzen. Das ist so, wie wenn den Zeitungen durch ein Gesetz verboten werden würde, auf fehlerhafte Arbeiten der Polizei hinzuweisen. Ein solches Gesetz schützt den Staat davor, sich Kritik aussetzen zu müssen. Dies ist der erste und wichtigste Schritt in die Richtung eines autoritären Systems.

6. Impressum

Herausgeber:

scip AG

Technoparkstrasse 1

CH-8005 Zürich

T +41 1 445 1818

<mailto:info@scip.ch>

<http://www.scip.ch>

Zuständige Person:

Marc Ruef

Security Consultant

T +41 1 445 1812

<mailto:maru@scip.ch>

PGP:

http://www.scip.ch/firma/facts/maru_scip_ch.asc

scip AG ist eine unabhängige Aktiengesellschaft mit Sitz in Zürich. Seit der Gründung im September 2002 fokussiert sich die scip AG auf Dienstleistungen im Bereich IT-Security. Unsere Kernkompetenz liegt dabei in der Überprüfung der implementierten Sicherheitsmassnahmen mittels **Penetration Tests** und **Security Audits** und der Sicherstellung zur Nachvollziehbarkeit möglicher Eingriffsversuche und Attacken (**Log-Management** und **Forensische Analysen**). Vor dem Zusammenschluss unseres spezialisierten Teams im Jahre 2002 waren die meisten Mitarbeiter mit der Implementierung von Sicherheitsinfrastrukturen beschäftigt. So verfügen wir über eine Reihe von Zertifizierungen (Solaris, Linux, Checkpoint, ISS, Okena, Finjan, TrendMicro, Symantec etc.), welche den Grundstein für unsere Projekte bilden. Den kleinsten Teil des Wissens über Penetration Test und Log-Management lernt man jedoch an Schulen – nur jahrelange Erfahrung kann ein lückenloses Aufdecken von Schwachstellen und die Nachvollziehbarkeit von Angriffsversuchen garantieren.

Nutzen Sie unsere Dienstleistungen!

Einem **konstruktiv-kritischen Feedback** gegenüber sind wir nicht abgeneigt. Denn nur durch angeregten Ideenaustausch sind Verbesserung möglich. Senden Sie Ihr Schreiben an smss-feedback@scip.ch.

Das [Errata](#) (Verbesserungen, Berichtigungen, Änderungen) der scip monthly Security Summary's finden Sie online.

Der Bezug des scip monthly Security Summary ist **kostenlos**. [Anmelden!](#) [Abmelden!](#)