CyberFortress Report

2020
JULY

CyberFortress

このレポートは、株式会社サイバーフォートレスの独自の調査にて収集されたデータに基づき作成しています。

webshellのパターン収集、診断スクリプトについて

01. 概要

webshellは簡単なサーバスクリプト(jsp, php, asp...)を活用してウェブページからサーバにコマンドを実行するために作られる。このようなスクリプトはサーバの管理者権限を獲得するため、ウェブサーバの脆弱性を利用してアップロードされる。

ウェブサービスが動作し、アップロードされていても脆弱性もしくは実行権限がないと実施されない。しかし、脆弱性と実行権限がある場合、サーバ内部からコマンドを実施し、情報の漏洩、ソースコードの確認、ページのお改ざんなどが行われる。

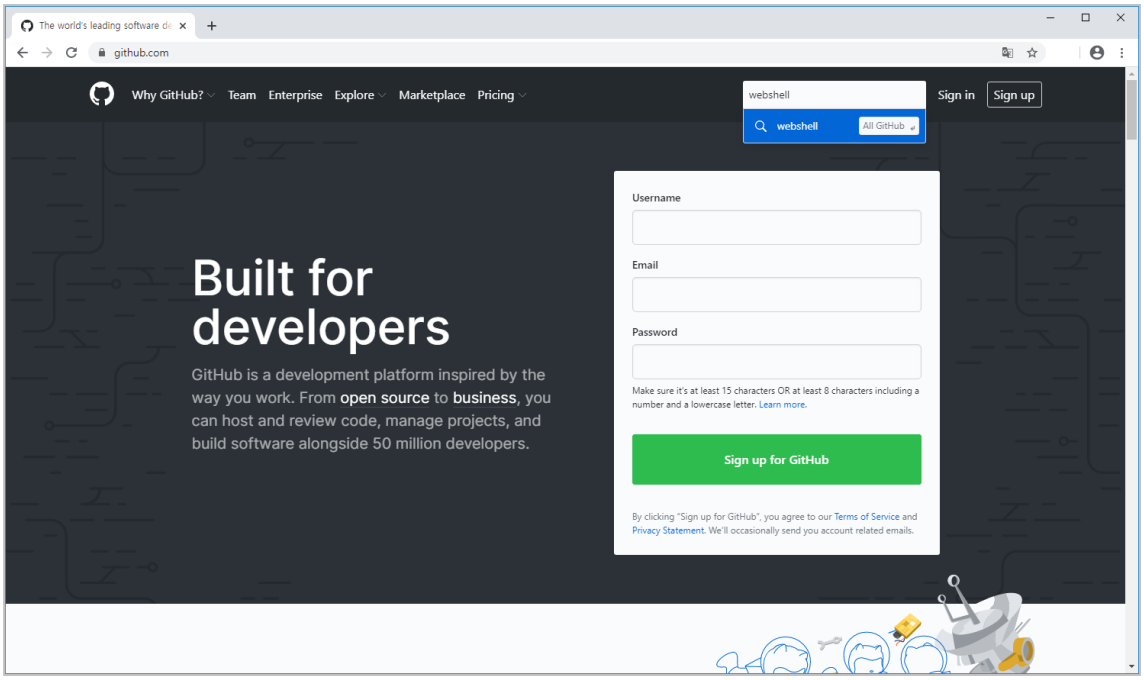
今回はwebshellを診断するためのパターンの収集方法と診断スクリプトの作成また、類型ごとの分析方法について調べてみよう。

02. webshell検索パターンの収集

1) webshellキーワード検索

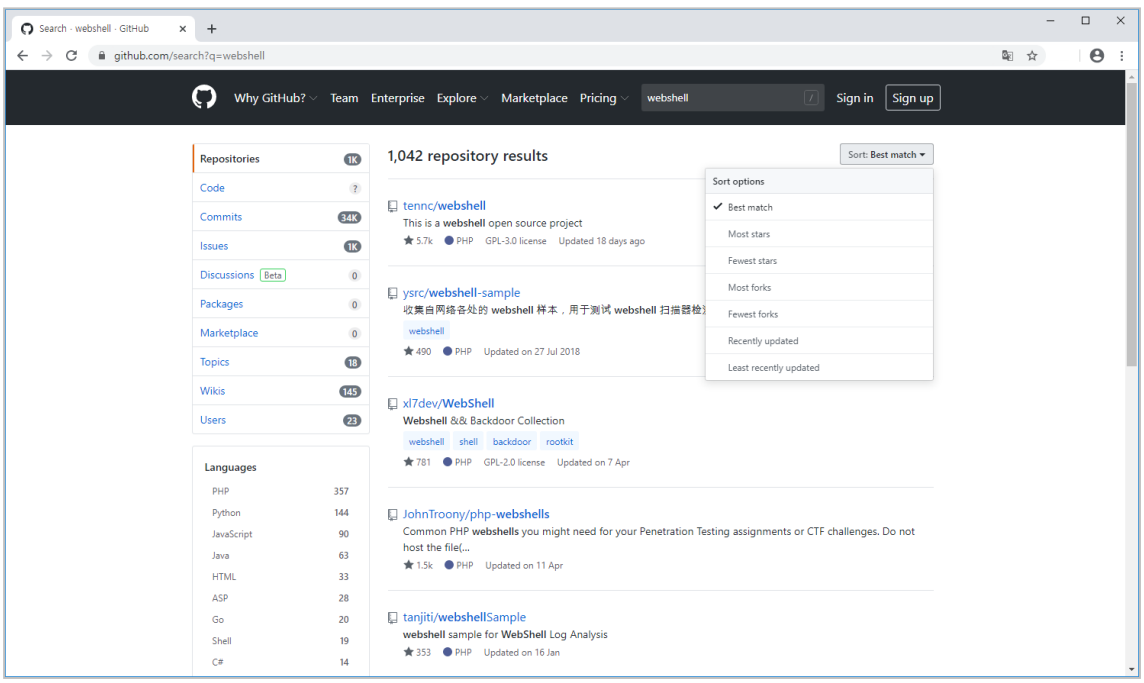
既にwebshellが判定出来ているのであれば、webshellのパターンを保有していると思っているがパターンがなく、webshellの判定が出来ない場合は、開発者向けオープンコミュニティ(https://github.com)にアクセスし、webshellのキーワードで検索する。

webshellのパターン収集、診断スクリプトについて



「Webshellのキーワードで検索」

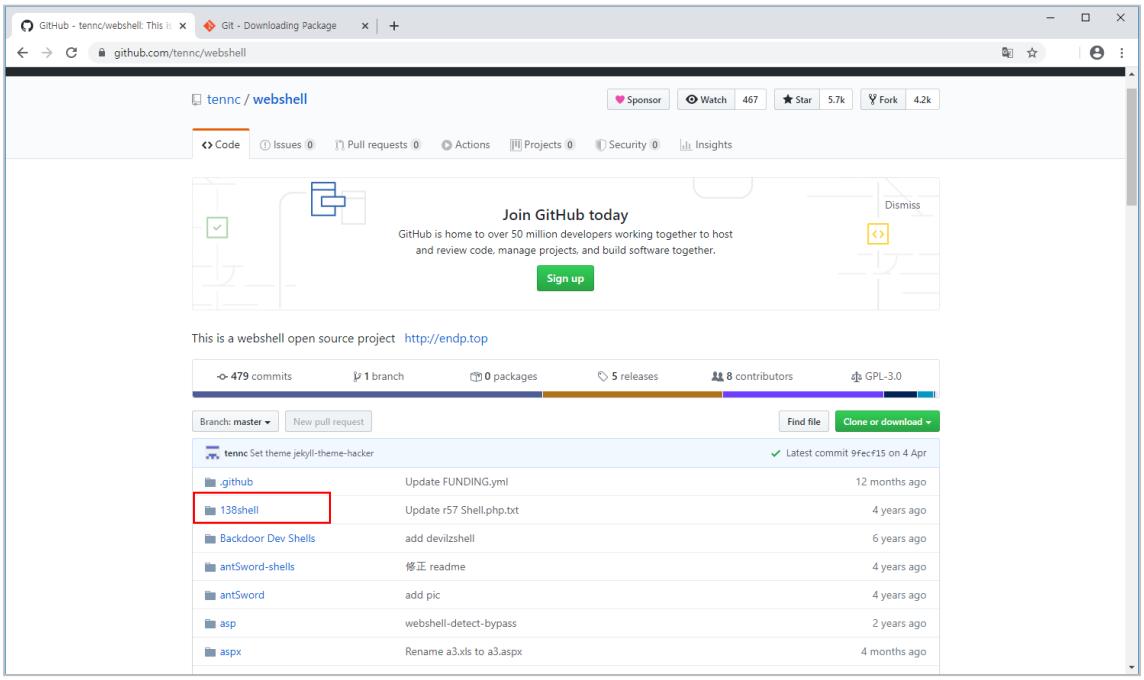
検索の結果はキーワードの単語に一番当てはまるBest matchな結果が表示される。



「webshellの検索結果確認」

webshellのパターン収集、診断スクリプトについて

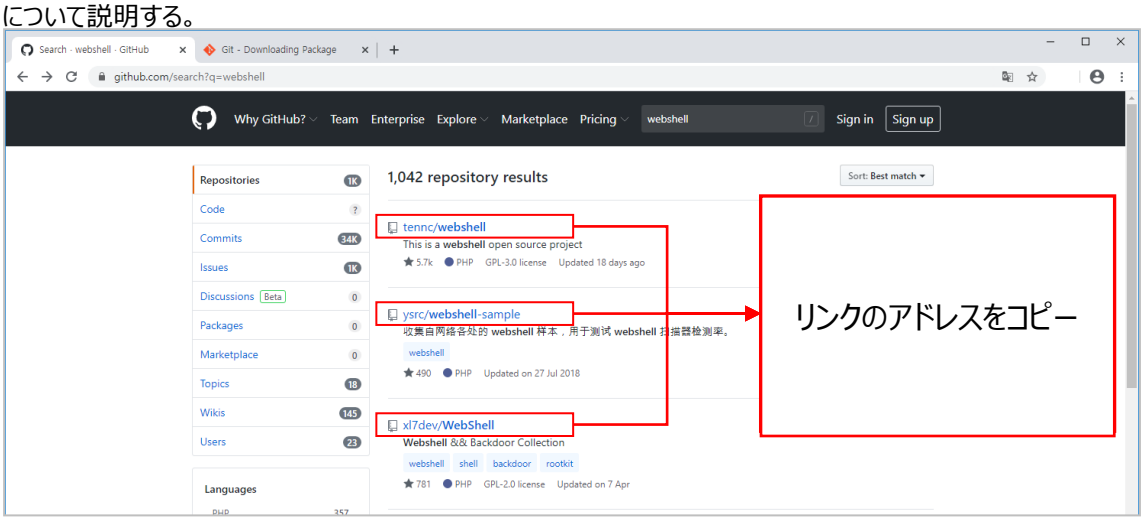
検索結果の中で一つのリンク(例：tennc / webshell)に直接アクセスし、ソースコードを確認し、コード内にwebshellとして判断できる文字列を確認する。



「ソースコード内にwebshellとして判断できる文字列を確認」

2) Gitツールの活用

それぞれのリンクからソースコードを確認し、またwebshellとして判断ができる文字列を確認するのは相当時間がかかる。Gitツールを活用することで複数のリンクを一括でダウンロードができるため、時間の短縮ができる。その方法について説明する。

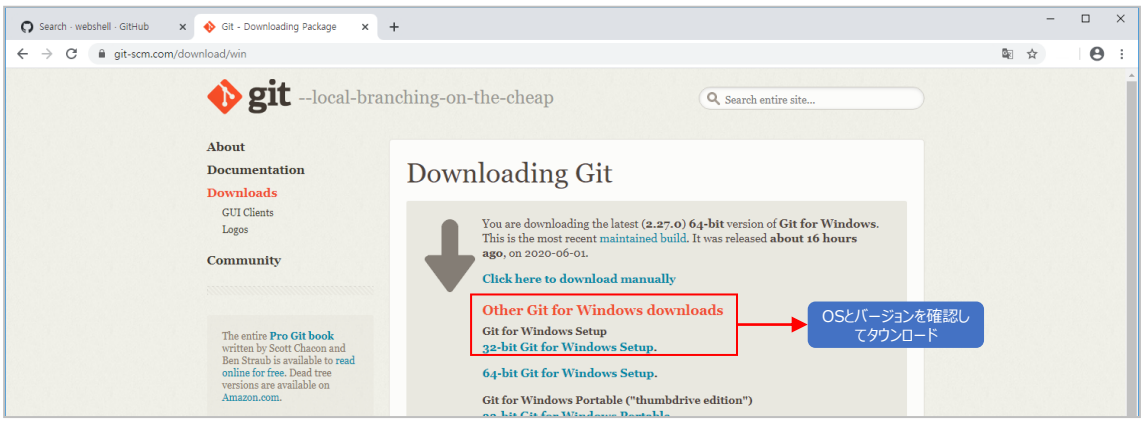


「リンクアドレスをコピー」

webshellのパターン収集、診断スクリプトについて

Gitツールは下記のURLからダウンロードできる。

< ダウンロードURL：https://git-scm.com/download/win >



「Gitツールのダウンロード方法」

Gitツールのインストールが終わったらコマンドについて調べてみよう。

① コピーしたリンクをメモ帳にgitコマンドの形式で保存
- コマンド：git clone「リンクURL」「保存するディレクトリ名」

```
git clone https://github.com/tennc/webshell tennc
git clone https://github.com/ysrc/webshell-sample ysrc
git clone https://github.com/xl7dev/WebShell xl7dev
git clone https://github.com/JohnTroony/php-webshells JohnTroony
```

「コピーしたリンクをGitコマンドの形式でメモ帳に保存」

② ツールがインストールされているディレクトリに移動し、メモ帳のコマンドを実行

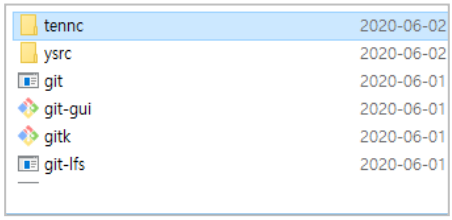
```
C:¥Program Files¥Git¥cmd>git clone https://github.com/tennc/webshell tennc
Cloning into 'tennc'...
remote: Enumerating objects: 57, done.
remote: Counting objects: 100% (57/57), done.
remote: Compressing objects: 100% (50/50), done.
remote: Total 2964 (delta 25), reused 0 (delta 0), pack-reused 2907
Receiving objects: 100% (2964/2964), 27.94 MiB | 7.76 MiB/s, done.
Resolving deltas: 100% (1156/1156), done.
Updating files: 100% (1595/1595), done.

C:¥Program Files¥Git¥cmd>git clone https://github.com/ysrc/webshell-sample ysrc
Cloning into 'ysrc'...
remote: Enumerating objects: 2789, done.
remote: Total 2789 (delta 0), reused 0 (delta 0), pack-reused 2789
Receiving objects: 100% (2789/2789), 35.82 MiB | 8.54 MiB/s, done.
Resolving deltas: 100% (406/406), done.
Updating files:  43% (1104/2566)
```

「メモ帳の内容を実行」

webshellのパターン収集、診断スクリプトについて

Gitコマンドを実行し、結果を確認すると複数のリンクがそれぞれのディレクトリに保存されていることが確認できる。



「Gitコマンドの結果」

3) Webshellパターンの抽出方法

メモ帳などを利用してソースコードを確認し、パターンとして追加できる文字列を手動で抽出し、 webshellとして開発される可能性が高いソースコードを選別して抽出する。

抽出の際、一般的にソースコード開発者が使わないと判断できるパターンを抽出する。抽出するパターンによって、webshellの診断スクリプトの正検知率と誤検知率に影響が与えられるので注意が必要である。

順位	パターンの類型	webshell検知パターンの例
1	知られているwebhshell名	cmdasp.asp, jFolder.jsp, alihack.phpなど
2	攻撃者が残した特定のシグネチャ、メールなど	h4cked, hacked by, by unknownなど
3	webhshellの機能ができる関数	Script.WShell, eval(gzinflate(base64_, echo exec(など
4	その他webshellとして判断ができる文字列	File Browser, filemanager, jfilemanなど

「webshellの類型ごと、検知パターン」

webshellのパターン収集、診断スクリプトについて

03. webshell診断スクリプト

webshell診断スクリプトは findコマンドを利用してサーバ内のファイルにwebshellのパターンを探すスクリプトである。今回はUNIX（LINUX）で使用できるスクリプトについて確認してみよう。

1) UNIX用スクリプト

① ファイルサイズは2048(1M)に設定し、診断対象は全てもファイル(*.*)に設定。

```
#!/bin/sh
# File Size Setting ( size=1024 -> 512K byte, size=2048 -> 1M byte, size=4096 -> 2M byte )
size=2048

# File Extension Setting (Default=all)
file_ext="*.*"
```

診断結果ファイルサイズを1M byteに設定

診断対象のファイルは全てのファイル(*.*)に設定

「ファイルサイズ設定と診断対象設定」

② 変数はcount、hostname、日付、時間、結果(3つ)を設定

```
# Global Variable Setting
LANG=C
export LANG
count=0
hostname=`hostname`
today=`date +%y%m%d`
totime=`date +%H%M%S`
sday=`date +%y.%m.%d`
stime=`date +%H:%M:%S`
result0=shell_"$hostname"
result1="$today"_"$totime"
result2="$hostname"_"$today"_"$totime"
```

「スクリプトから使用する変数設定」

③ if文を使用し、webshellパタンファイルに対して自動圧縮解凍スクリプトを設定

```
if [ -f usig.tar.gz ]
  then
    gzip -d usig.tar.gz
    tar -xvf usig.tar
else
    echo "Not Found File usig.tar.gz!!"
    exit 0
fi
```

「webshellパターン自動圧縮解凍」

webshellのパターン収集、診断スクリプトについて

④ fileディレクトリを作成し、webshell診断結果を保存するように設定

```
53  # Create Folder (Result File & Web shell File)
54  mkdir -p ./"$result0"/"$result1"/file
```

「webshellパターン診断結果保存」

⑤ 診断対象の情報を結果ファイルに(result_log.tsv)に出力できるように設定。

```
56  # Script Start Info
57  clear
58  echo "" 2>&1
59  echo " Web Shell Find Script File" 2>&1
60  echo "" 2>&1
61  echo " - Start Time: $sday $stime" 2>&1
62  echo "Start Time: $sday $stime">./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
63
64  echo "Hostname: $hostname" >> ./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
65  ifconfig | grep "inet addr" >> ./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
66
67  echo " - Directory : $1" 2>&1
68  echo "Directory : $1">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
69  echo " - File Size : $size" 2>&1
70  echo "File Size : $size">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
71  echo " - File Extension: $file_ext" 2>&1
72  echo "File Extension: $file_ext">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
73  echo "" 2>&1
74  echo "  File Listing..." 2>&1
75  echo "" 2>&1
76  echo "Webshell Copy File List">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
```

「result_log.tsvの出力情報」

⑥ webshellパターン文字列を検索し、マッチングされた結果を「webshell_list.tsv」ファイルに保存されるように設定

```
78  # Webshell Find
79  find "$1" -type f -exec ls -Ralt {} \;  >> ./"$result0"/"$result1"/"$result2"_file_list.tsv 2>&1
80  find "$1" -type f -size -"$size" -name "$file_ext" -exec fgrep -l -i -f usig /dev/null {} \; >> ./"$result0"/
    "$result1"/"$result2"_webshell_list.tsv 2>&1
```

「webshellパターン文字列の検索及びファイル作成」

⑦ webshell検索結果からファイルパスを整理し、当該のファイルをtxtファイルとして作成されるように設定

```
85  # Webshell File Copy
86  for filename in `cat ./"$result0"/"$result1"/"$result2"_webshell_list.tsv`
87  do
88      count=`expr $count + 1`
89      cp -p "$filename" ./"$result0"/"$result1"/file/$count.txt
90      echo "$count.txt = $filename">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
91  done
92
93  file_num=`find ./"$result0"/"$result1"/file/ -type f | wc | awk '{ print $1}'`
94  file_list=`cat ./"$result0"/"$result1"/"$result2"_file_list.tsv | wc | awk '{ print $1}'`
95  file_512=`find "$1" -type f -size -"$size" | wc | awk '{ print $1}'`
96  copy_dir_size=`du ./"$result0"/"$result1"/file/ | awk '{print $1}'`
```

「webshellの結果をtxtファイルとして作成」

⑧ 結果ファイル作成(result_log.tsv)

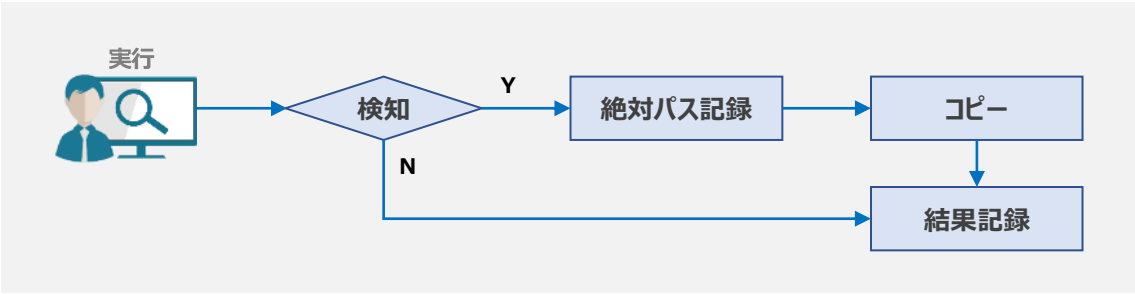
```
98  echo """>>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
99  echo " - File Count: $file_512 files [all $file_list files]" 2>&1
100 echo "File Count: $file_512 files [all $file_list files]">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
101 echo " - File Copy : $file_num files [$copy_dir_size bytes]" 2>&1
102 echo "File Copy : $file_num files [$copy_dir_size bytes]">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
103 echo " - Result Dir: $result0/$result1" 2>&1
104 echo "Result Dir: $result0/$result1">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
105 echo " - End Time  : `date +%y.%m.%d` `date +%H:%M:%S`" 2>&1
106 echo "End Time  : `date +%y.%m.%d` `date +%H:%M:%S`">>./"$result0"/"$result1"/"$result2"_result_log.tsv 2>&1
```

「結果ファイル作成」

webshellのパターン収集、診断スクリプトについて

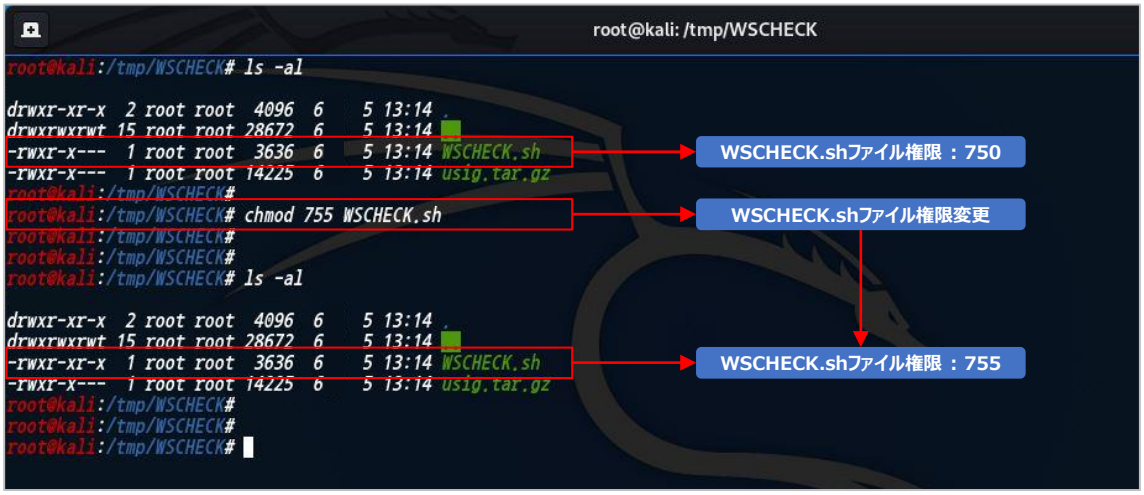
2) 診断スクリプトの実行方法

診断スクリプトのパラメータに診断するディレクトリパスを入力し実行させる。当該のパスの下位に存在する全てのファイルの内容(ソースコード)を分析し、webshellパターンが含まれているか確認する。
Webshellが検知されたら絶対パスを記録し、コピーし結果を記録しよう。



「webshell診断スクリプトのプロセス」

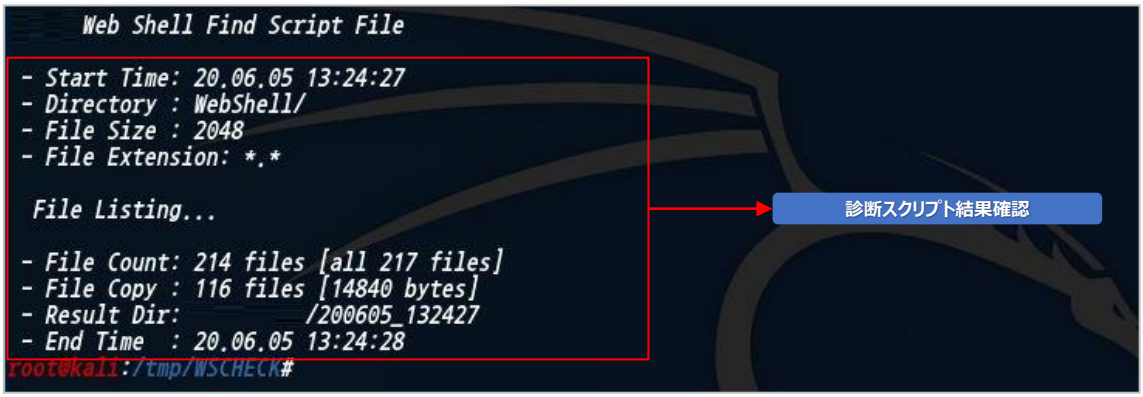
Webshell診断スクリプトファイルと、パターンが保存されているファイルの2つを、検索されたwebshellのコピーファイルが保存できるようにスクリプトの権限、保存場所のディスクスペースを確認してから実施する。



「webshellスクリプトの権限変更」

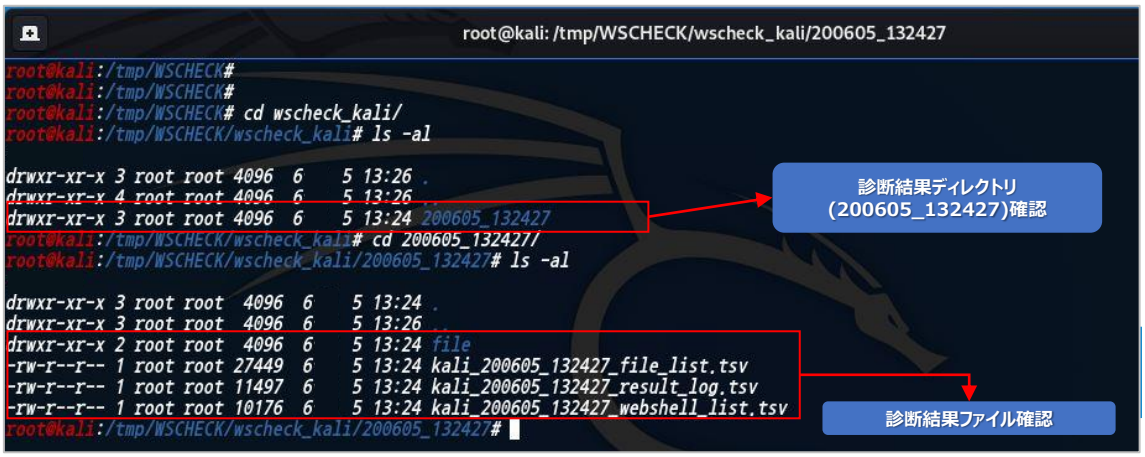
「診断対象のディレクトリを指定し、スクリプト実行」

webshellのパターン収集、診断スクリプトについて



「webshellスクリプト結果確認」

診断が終わるとスクリプトが実行されたディレクトリに３つの診断結果ファイルが作成される。



「webshell診断後、作成されたディレクトリ及びファイル」

順位	ディレクトリ及びファイル名	説明
1	file	検知されたwebshellのコピーを「数字.txt」形式で保存
2	file_list.tsv	診断対象のディレクトリにある全てのファイル情報を保存(下位ディレクトリを含め)
3	result_log.tsv	スクリプトの全般的な情報を保存
4	webshell_list.tsv	検知されたwebshellのパス及びファイル情報を保存

「webshell診断結果ファイルの説明」

webshellのパターン収集、診断スクリプトについて

04. Webshell類型ごと分析方法

1) Webshellとして判断できる対象

ソースコード内にファイルの拡張子が確認された場合、もしくは各種のfile browser形式のソースコードの場合、拡張子がなくてもwebshellとして判断ができ、検知されたファイルの中に「.as%70, .js%70, ;.jpg, ;.gif」のように拡張子の部分に記号などが含まれている異常のファイルも webshellとして判断する。
異常ファイルのwebshell区別が不明であればwebshellと同じだと判断し削除が必要である。

```
/project/app/ons/bmsWeb/bms/agd/BmsAgdAInfoListExcel.jsp
/project/app/ons/bmsWeb/bms/com/BmsMyDesk03.jsp
/project/app/ons/bmsWeb/bms/com/BmsMyDesk066.jsp
/project/app/ons/bmsWeb/bms/com/2852cmd.jsp
/project/app/ons/bmsWeb/bms/com/3521cmd.jsp
/project/app/ons/bmsWeb/bms/com/3985cmd.js%70
/project/app/ons/bmsWeb/bms/com/4412cmd.js%70
/project/app/ons/bmsWeb/bms/com/8490cmd.jpg
/project/app/ons/bmsWeb/bms/ord/BmsOrdCardList02L.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdCardListHadal.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdDisUseInstPreviewList.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdInstL.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdInstListPortlet.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdInstP.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdInstResultOnlyV.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdInstResultV.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdInstV.jsp
/project/app/ons/bmsWeb/bms/ord/BmsOrdOgmInstList.jsp
```

「日付または時間の形式で作成されているファイル名」

```
<%--
    jsp File browser 1.0
    Copyright (C) 2003,2004, Boris von Loesch
    This program is free software; you can redistribute it and/or modify it under
    the terms of the GNU General Public License as published by the
    Free Software Foundation; either version 2 of the License, or (at your option)
    any later version.
    This program is distributed in the hope that it will be useful, but
    WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
    FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
    You should have received a copy of the GNU General Public License along with
    this program; if not, write to the
    Free Software Foundation, Inc.,
    59 Temple Place, Suite 330,
    Boston, MA 02111-1307 USA
    - Description: jsp File browser v1.0 -- This JSP program allows remote web-based
                   file access and manipulation.  You can copy, create, move and delete files.
                   Text files can be edited and groups of files and folders can be downloaded
                   as a single zip file that's created on the fly.
    - Credits: Taylor Bastien, David Levine, David Cowan
--%>
<%@page import="java.util.*,
                java.net.*,
                java.text.*,
                java.util.zip.*,
                java.io.*" %>
```

jsp File browserのソースコード

webshellのパターン収集、診断スクリプトについて

2) 単純ソースコードとして判断ができる対象

イメージファイルの拡張子(.jpg，.gif，.bmp，.pngなど)を除き、ほかの拡張子のファイル名、形式、またはファイル名の長さが同じ場合、APの開発の中で履歴を管理する目的で作成されたファイルであるため、単純パターンマッチングとして判断できる。

```
-rwxr-x---    1 weblogic swgrp          1113 Mar 14 2019  /project/app/WebApp/backup/manual.m
-rwxr-x---    1 weblogic swgrp         52661 Mar 14 2019  /project/app/WebApp/backup/http.m
-rwxr-x---    1 weblogic swgrp           909 Oct  5 2019  /project/app/WebApp/backup/http.m.20191005
-rwxr-x---    1 weblogic swgrp         13322 Dec  3 2019  /project/app/WebApp/backup/http.m.20191203
-rwxr-x---    1 weblogic swgrp          2821 Oct 31 2019  /project/app/WebApp/backup/20191005.xml
-rwxr-x---    1 weblogic swgrp          6857 Dec  3 2019  /project/app/WebApp/backup/wsconfig.20191203
-rwxr-x---    1 weblogic swgrp          1194 Mar 14 2019  /project/app/WebApp/backup/http.m.bak
-rwxr-x---    1 weblogic swgrp          6857 Oct  5 2019  /project/app/WebApp/backup/wsconfig.20191005
-rwxr-x---    1 weblogic swgrp         13622 Mar 14 2019  /project/app/WebApp/backup/wsconfig
```

「日付または時間の形式のファイル名」

ウェブコンテナ(jeus，weblogic，tomcatなど)から共通で使用したり、各種アプリケーション(java，oracle，openssなど)がインストールされているパスに存在している基本ソースコードもしくは、コンパイルされたファイル(.c，.h，.java，.classなど)などはウェブサービスに必要なファイルで、webshellではないと判断できるが、オープンソースの場合は内容を確認して直接判断が必要である。

```
-rw-rw-r--    1 weblogic swgrp          2638 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppBisBasicViewCmd.class
-rw-rw-r--    1 weblogic swgrp          2639 Mar 31 2018  /project/app/WebApp/WEB-INF/classes//AppBisLevelViewCmd.class
-rw-rw-r--    1 weblogic swgrp          2701 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppBisSmlViewCmd.class
-rw-rw-r--    1 weblogic swgrp          2905 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppDetailListCmd.class
-rw-rw-r--    1 weblogic swgrp          2408 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppHisDetailListCmd.class
-rw-rw-r--    1 weblogic swgrp          2857 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppPjtDetailViewCmd.class
-rw-rw-r--    1 weblogic swgrp          3154 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppReqCancelUpdCmd.class
-rw-r--r--    1 weblogic swgrp          4647 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppReqListCmd.class
-rw-rw-r--    1 weblogic swgrp          4595 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppReqListCmd.class_061220
-rw-rw-r--    1 weblogic swgrp          2473 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppReqResultViewCmd.class
-rw-rw-r--    1 weblogic swgrp          3112 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppResAgreeCmd.class
-rw-r--r--    1 weblogic swgrp          4651 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppResListCmd.class
-rw-rw-r--    1 weblogic swgrp          4549 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppResListCmd.class_061220
-rw-rw-r--    1 weblogic swgrp          2473 Mar 31 2018  /project/app/WebApp/WEB-INF/classes/AppResResultViewCmd.class
```

「基本ソースコードもしくは、コンパイルされたファイル名」

Webshellで使用されている文字列がマッチしたが、日本語で注釈などがある場合、サービスの目的で作成されたファイルとして判断できる。

webshellのパターン収集、診断スクリプトについて

```
<%@ Language=VBScript %>
<!--#include virtual="/_include/logincheck.asp"-->
<!--#include virtual="/_include/open.asp"-->
<!--#include virtual="/_include/common_function.inc"-->

<%
    Set listRS = Server.CreateObject("ADODB.Recordset")
    listRS.CursorLocation = 3

    Dim Rows        ``一つのページに見せる投稿数
    Dim total_pages ``全てのページの数
    Dim total_record``全ての投稿の数
    dim curpage

    Rows = 10
    total_pages = 0
    total_record = 0

    curpage = Request("curpage")
    idx = Request("hope_idx")

    'stype      = Trim(Request("searchtype"))
    'sword      = Trim(Request("searchword"))

    url = Request.ServerVariables("URL")
    page = mid(url, InstrRev(Request.ServerVariables("URL"), "/", -1, 1)+1)
    linkpage = page
```

日本語で注釈された場合、
使用しているAPソースとして判断

webshell の文字列
(Request.ServerVariables)

「日本語で注釈された基本ソースコード」

05. 結論

Webshellパターンはウェブサイトまたはインシデント対応の中で収集できるが、診断スクリプトを作成すのはLinux Shell Scriptの理解が必要であるため、十分に内部テストの後に本番サーバで実施することを推奨する。

インシデント対応の立場ではセキュリティ機器からwebshellを検知及び遮断するためのポリシーを開発するのは限界がある。そのため、周期的にwebshell診断を行ったり、稼働されているサーバが少ない場合、cronを利用して周期的に診断できるように設定することを推奨する。