



Final Engagement

Attack of a Vulnerable Network

Table of Contents

This document contains the following resources:

01

**Network Topology &
Vulnerabilities**

02

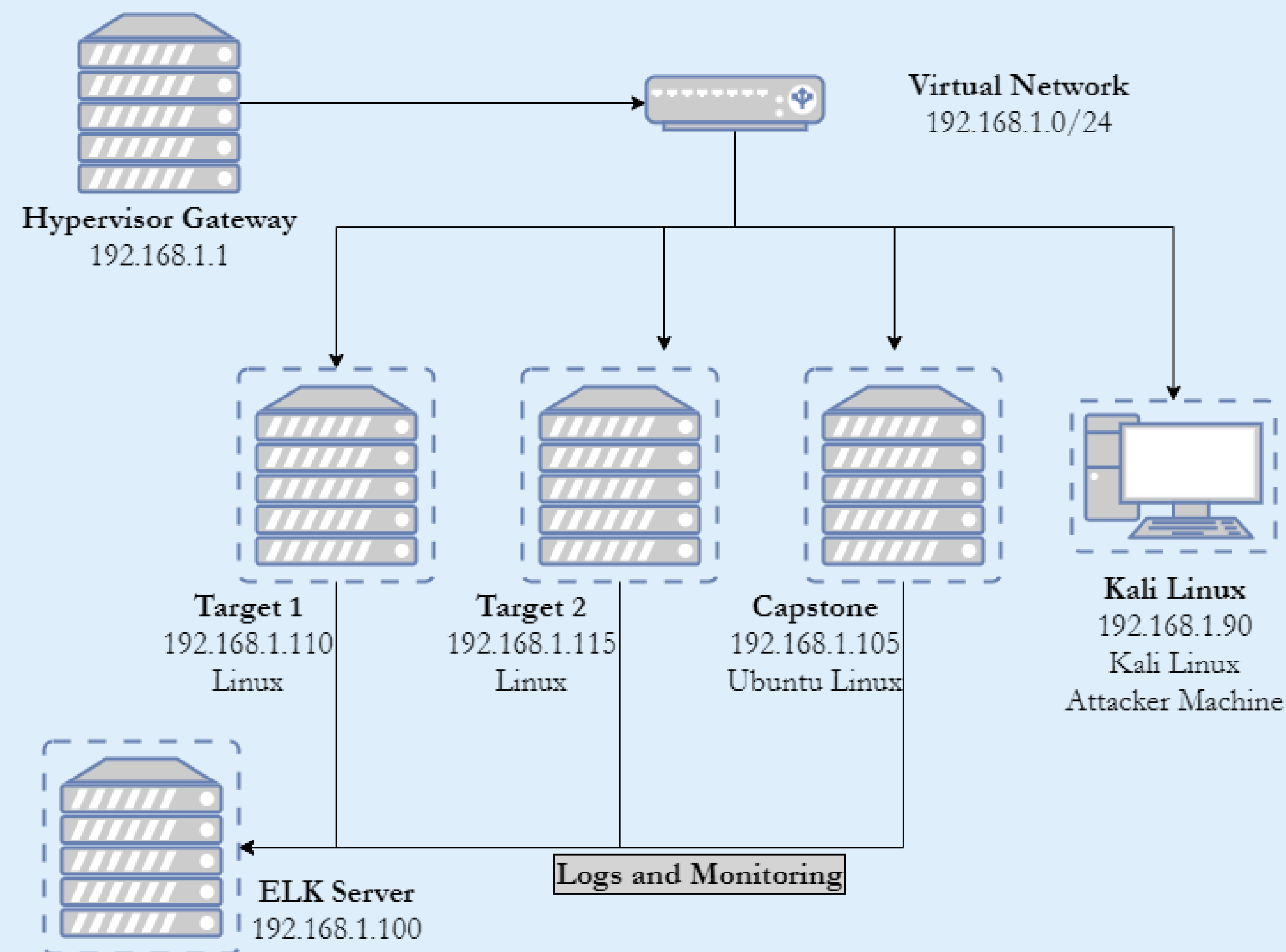
Exploits Used

03

Mitigations

Network Topology & Vulnerabilities

Network Topology



Network

Address Range:
192.168.1.0/24
Netmask:
255.255.255.0
Gateway: 192.168.1.1

Machines

IPv4: 192.168.1.90
OS: Debian Kali 5.4.0
Hostname: Kali

IPv4: 192.168.1.100
OS: Ubuntu Linux 18.04
Hostname: ELK

IPv4: 192.168.1.105
OS: Ubuntu Linux 18.04
Hostname: Capstone

IPv4: 192.168.1.110
OS: Debian GNU/Linux 8
Hostname: Target 1

Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
Common web dir naming.	Having the wordpress software in a folder rather than the http root/vhost.	Allows wordpress to be more easily identified as the framework in use.
Incorrect dir permissions.	Enabling a user other than the owner/webserver to access dirs.	Allows users to view or modify files/folders containing sensitive information
SUID bit set for python.	Allows users to run python and therefore code as root	Privilege escalation to root.
Root database user.	Using the root user access for a web app rather than a specific credentials.	Malicious user/application access to entire database.

Exploits Used

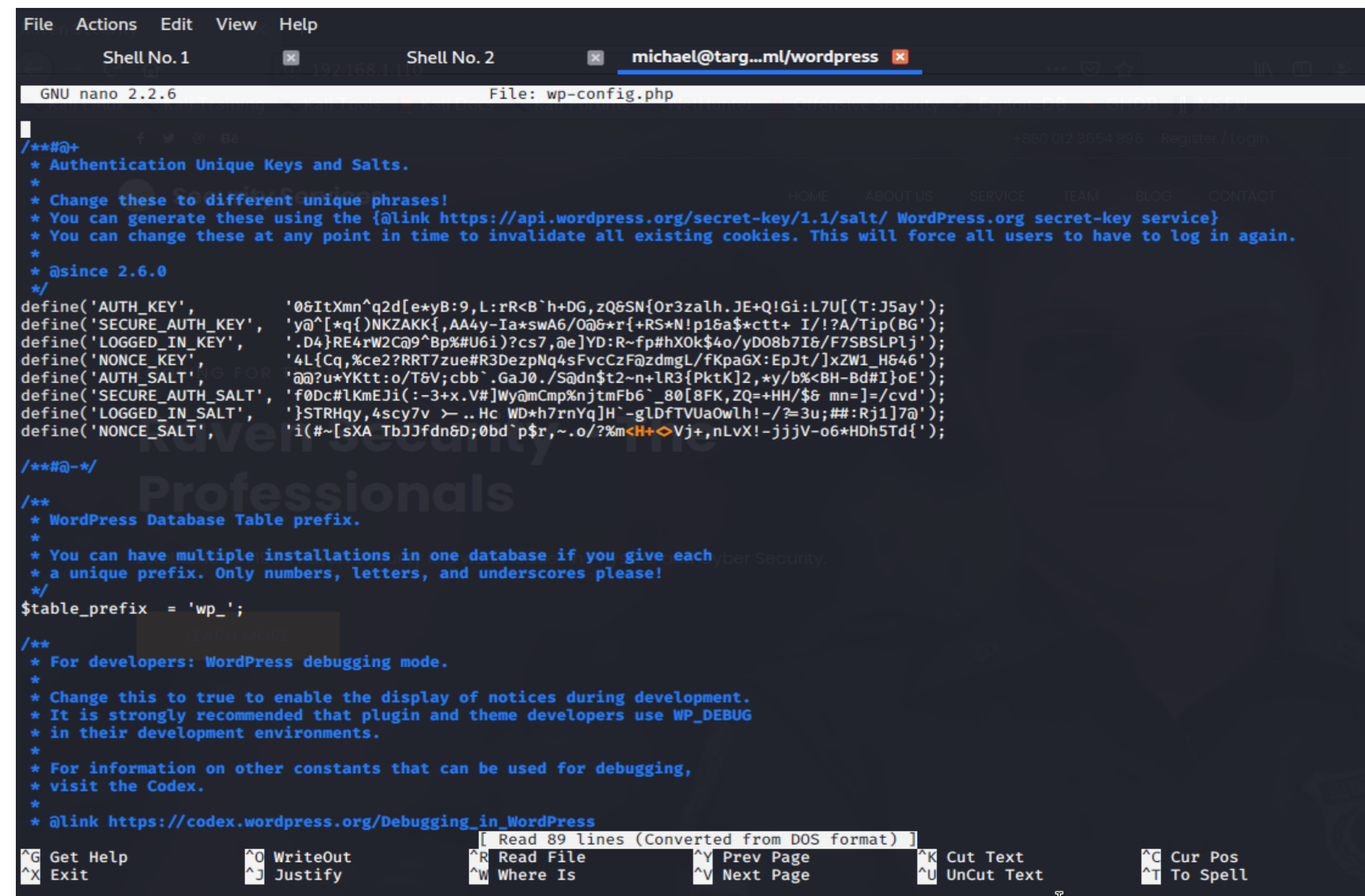
Exploitation: Common web dir naming

- Tool: dirb
- Outcome: Located the /wordpress/ directory on the target, identifying the web framework in use.

```
--> Testing: http://192.168.1.110/word
--> Testing: http://192.168.1.110/wordpress
==> DIRECTORY: http://192.168.1.110/wordpress/
```


Exploitation: Incorrect dir permissions

- Tool: ls -la
- Outcome: The wordpress directory permissions were incorrectly set allowing all users to read the configuration file for the wordpress installation which contains sensitive information including database credentials in plaintext .



The screenshot shows a terminal window with a nano editor open to the file wp-config.php. The editor's title bar indicates the user is michael@targ...ml/wordpress. The file content is a PHP configuration script for WordPress, starting with a comment block about authentication keys and salts. It defines several constants for keys and salts, including AUTH_KEY, SECURE_AUTH_KEY, LOGGED_IN_KEY, NONCE_KEY, AUTH_SALT, SECURE_AUTH_SALT, LOGGED_IN_SALT, and NONCE_SALT. It also defines the database table prefix as 'wp_'. The bottom of the screen shows a status bar with various keyboard shortcuts like Get Help, WriteOut, Read File, Justify, Where Is, Prev Page, Next Page, Cut Text, UnCut Text, Cur Pos, and To Spell.

```
File Actions Edit View Help
Shell No. 1 Shell No. 2 michael@targ...ml/wordpress
GNU nano 2.2.6 File: wp-config.php

/**#@+
 * Authentication Unique Keys and Salts.
 *
 * Change these to different unique phrases!
 * You can generate these using the {link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}
 * You can change these at any point in time to invalidate all existing cookies. This will force all users to have to log in again.
 *
 * @since 2.6.0
 */
define('AUTH_KEY', '0&ItXmn^q2d[e*yB:9,L:rR<B`h+DG,zQ&SN{0r3zalh.JE+Q!Gi:L7U[(T:J5ay');
define('SECURE_AUTH_KEY', 'y@^[*q{)NKZAKK{,AA4y-Ia*swA6/0@&r{+RS*N!p1&a$*ctt+ I/!?A/Tip(BG');
define('LOGGED_IN_KEY', '.D4}RE4rW2C@9^Bp%#U6i)?cs7,@e]YD:R~fp#hX0k$4o/yD08b7I&/F7SBSLPlj');
define('NONCE_KEY', '4L{Cq,%ce2?RRT7zue#R3DezpNq4sFvcCzF@zdmgL/fKpaGX:EpJt/]xZW1_H&46');
define('AUTH_SALT', '@@?u*YKtt:o/T&V;cbB`.GaJ0./S@dn$t2~n+LR3{PktK]2,*y/b%<BH-Bd#I)oE');
define('SECURE_AUTH_SALT', 'f0Dc#lKmEji(:-3+x.V#]Wy@mCmp%njtmFb6`_80[8FK,ZQ=+HH/$& mn=]/cvd');
define('LOGGED_IN_SALT', 'STRHqy,4scy7v >..Hc WD*h7rnYq]H`-gLDfTVUaOwLh!-/=?3u;##:Rj1]7@');
define('NONCE_SALT', 'i(#~[sXA TbJJfdn&D;0bd`p$r,~.o/?%m<H+>Vj+,nLvX!-jjjV-o6*HDh5Td{');

/**#@-*/

/**
 * WordPress Database Table prefix.
 *
 * You can have multiple installations in one database if you give each
 * a unique prefix. Only numbers, letters, and underscores please!
 */
$table_prefix = 'wp_';

/**
 * For developers: WordPress debugging mode.
 *
 * Change this to true to enable the display of notices during development.
 * It is strongly recommended that plugin and theme developers use WP_DEBUG
 * in their development environments.
 *
 * For information on other constants that can be used for debugging,
 * visit the Codex.
 *
 * @link https://codex.wordpress.org/Debugging_in_WordPress
 */
define('WP_DEBUG', false);
```


Exploitation: SUID bit set for python

Summarize the following:

- Tool: sudo -l
- Outcome: Displays applications the user is able to run under sudo privileges, in this case python

```
Shell No.1  michael@targ...ml/wordpress
/usr/bin/newgrp
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/sudo
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmccrypt-get-device
/usr/sbin/sensible-mda
/sbin/mount.nfs
/sbin/mount.cifs
michael@target1:/var/www/html/wordpress$ su steven
Password:
$ ls -la
total 668
drwxrwxrwx  5 root    root      4096 Oct 27 23:11 .
drwxrwxrwx 10 root    root      4096 Aug 13  2018 ..
-rw-r--r--  1 www-data www-data  255 Aug 13  2018 .htaccess
-rwxrwxrwx  1 root    root       418 Sep 25  2013 index.php
-rwxrwxrwx  1 root    root     19935 Aug 13  2018 license.txt
-rwxrwxrwx  1 root    root      7413 Oct 27 22:43 readme.html
-rw-r--r--  1 michael michael 473987 Oct 27 23:11 wordpress.sql
-rwxrwxrwx  1 root    root     6864 Oct 27 22:43 wp-activate.php
drwxrwxrwx  9 root    root      4096 Jun 15  2017 wp-admin
-rwxrwxrwx  1 root    root       364 Dec 19  2015 wp-blog-header.php
-rwxrwxrwx  1 root    root     1627 Aug 29  2016 wp-comments-post.php
-rw-rw-rw-  1 www-data www-data  3134 Aug 13  2018 wp-config.php
-rwxrwxrwx  1 root    root     2853 Dec 16  2015 wp-config-sample.php
drwxrwxrwx  6 root    root      4096 Oct 27 22:44 wp-content
-rwxrwxrwx  1 root    root     3286 May 24  2015 wp-cron.php
drwxrwxrwx 18 root    root    12288 Jun 15  2017 wp-includes
-rwxrwxrwx  1 root    root     2422 Nov 21  2016 wp-links-opml.php
-rwxrwxrwx  1 root    root     3301 Oct 25  2016 wp-load.php
-rwxrwxrwx  1 root    root    34347 Oct 27 22:43 wp-login.php
-rwxrwxrwx  1 root    root     8048 Jan 11  2017 wp-mail.php
-rwxrwxrwx  1 root    root    16200 Apr  6  2017 wp-settings.php
-rwxrwxrwx  1 root    root    29924 Jan 24  2017 wp-signup.php
-rwxrwxrwx  1 root    root     4513 Oct 14  2016 wp-trackback.php
-rwxrwxrwx  1 root    root     3065 Aug 31  2016 xmlrpc.php
$ sudo -l
Matching Defaults entries for steven on raven:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin
User steven may run the following commands on raven:
  (ALL) NOPASSWD: /usr/bin/python
$
```

Exploitation: Root database user

- Tool: mysql
- Outcome: By exposing the root user credentials, we were able to dump all the records from the database including user credentials for other areas.



```
Shell No. 1      Shell No. 2      michael@targ...ml/wordpress
GNU nano 2.2.6      File: wp-config.php

?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://codex.wordpress.org/Editing_wp-config.php
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'R@v3nSecurity');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8mb4');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');

/**#@+
 * Read 89 lines (Converted from DOS format)
 *
 * ^G Get Help      ^O WriteOut      ^R Read File      ^Y Prev Page      ^K Cut Text      ^G Cur Pos
 * ^X Exit          ^J Justify       ^W Where Is      ^V Next Page     ^U UnCut Text    ^T To Spell
 */
```

Exploitation Mitigations

Exploitation of Common web directory naming

Mitigation

- Set monitoring alerts for above normal numbers 404 errors.
- Set monitoring alerts for above normal file requests/minute.
- Consider a virtual host or installing to the web root.

Exploitation of Incorrect directory permissions

Mitigation

- Utilise the least permissions practice.
- Create a user or group for specific access.
- If possible, do not store plain text credentials. Nb in the case of wordpress and many PHP webapps this is just not possible.

Exploitation of SUID bit set for python

Mitigation

- Utilise the least permissions practice.
- Harden filesystem access preventing SUID use.

Exploitation of Root database user

Mitigation

- Ensure use of a highly secure root password.
- Give each webapp it's own database credentials and grant access only to it's individual database or tables.
- If possible do not store credentials in plain text.