

Monitorizacion WUX

Introducción

Pandora WUXServer es un componente interno del servidor Pandora FMS, que permite a los usuarios automatizar sus sesiones de navegación web.

Para poder funcionar, WUXServer deberá conectarse a un servidor PWRD (Pandora Web Robot Daemon).

PWRD es un servicio que ejecuta paso a paso acciones pre-grabadas por el usuario en un portal web.

Para lograr esto, PWRD utiliza por debajo la tecnología de automatización web de Selenium.

Será por tanto imprescindible, que el usuario indique a Pandora WUXServer las acciones que se deben realizar para navegar por su portal web.

Con estos datos, Pandora WUXServer generará un informe con los resultados de las ejecuciones, tiempos empleados, y capturas con los posibles errores encontrados.

Pandora WUXServer Es capaz de dividir las sesiones de navegación en fases, lo que simplificará la vista y nos ayudará a depurar posibles cuellos de botella.

Configuración

Para poder habilitar WUXServer en nuestro servidor Pandora FMS deberemos configurar el servicio **Pandora Web Robot (PWRD)**

Qué es Pandora Web Robot Daemon (PWRD)

Pandora Web Robot Daemon es un servicio que provee las herramientas necesarias para automatizar las sesiones de navegación web, utilizando un archivo que contiene una lista de las acciones necesarias para navegar por el portal web del usuario.

Viene integrado con el servidor de Pandora y lo podemos encontrar en la carpeta `/usr/share/pandora_server/util/pwr` al instalar el servidor Enterprise (Linux) o en la librería de módulos (Windows).

Contiene:

- Binario del navegador Firefox versión 46
- Perfil preconstruído para la ejecución de sesiones de navegación web
- Servidor de automatización de sesiones (Selenium server)

Despliegue de PWRD en Linux

El componente Selenium del demonio PWRD requiere java para funcionar, necesitaremos instalarlo:

```
yum install java
```

Para que PWRD sea capaz de lanzar Firefox en nuestro servidor Linux, será necesario instalar `xorg-x11-server-Xvfb` para poder virtualizar un entorno gráfico:

```
yum install xorg-x11-server-Xvfb
```

Si no está disponible en sus repositorios puede encontrar el fichero rpm en el siguiente enlace: http://rpmfind.net/linux/centos/6.6/os/x86_64/Packages/xorg-x11-server-Xvfb-1.15.0-22.el6.centos.x86_64.rpm

Para realizar la instalación del paquete rpm de forma manual:

```
yum install xorg-x11-server-Xvfb-1.15.0-22.el6.centos.x86_64.rpm
```

Una vez cumplidas las dependencias, procederemos a instalar el demonio PWRD, lo podemos encontrar en la carpeta `/usr/share/pandora_server/util/pwr` al instalar el servidor enterprise:

```
sudo /bin/bash /usr/share/pandora_server/util/install_pwr.sh --install
```

Una vez instalado, deberemos iniciarlo.

PWRD provee varios modos de funcionamiento:

- **Standalone:** Modo estándar, iniciará una única instancia de PWRD. Cada una de estas instancias irá asociada a un servidor de Pandora
- **HUB:** Modo concentrador. En este modo el servicio PWRD no evaluará las sesiones de navegación directamente, sino que deberán registrarse "nodos" para ejecutar las tareas. Es el modo clúster del servicio PWRD. Cada HUB irá asignado a un servidor de Pandora.

PWRD en modo standalone

PWRD en modo standalone, iniciará el demonio y lo dejará listo para ejecutar las acciones indicadas por el usuario a través de WUXServer.

```
# Iniciar
/etc/init.d/pwrld start

# ver el estado
/etc/init.d/pwrld status

# detener
/etc/init.d/pwrld stop
```

PWRD en modo "HUB"

El modo concentrador (o HUB) iniciará el demonio como un balanceador de carga. En este modo de trabajo, el sistema irá balanceando la carga entre todos los nodos que se hayan registrado en él, asignando la ejecución de las sesiones de navegación a los nodos en función de su carga de trabajo.

Puede ver en todo momento el estado del HUB accediendo a la consola de administración del mismo:

<http://hub:4444/grid/console>

```
# Iniciar
/etc/init.d/pwrld start-hub

# ver el estado
/etc/init.d/pwrld status-hub

# detener
/etc/init.d/pwrld stop-hub
```

Agregar nodos de PWRD a nuestro HUB

Para agregar un nuevo nodo PWRD, necesitaremos:

- Un concentrador (PWRD en modo HUB)
- Los archivos de PWRD, en la misma o una máquina diferente
- Conectividad TCP/4444 desde el equipo que aloja el nodo hacia el equipo que aloja el HUB.

En este modo de trabajo, el servicio procesará todas aquellas solicitudes encoladas **desde el concentrador (HUB)**, devolviendo a este los resultados de las ejecuciones. Será el HUB quien hable de manera exclusiva con el WUXServer, siendo transparente para este quién esté ejecutando las acciones del usuario.

```
# Iniciar y registrar en el HUB, sustituir "hub" por la IP del servidor PWRD HUB
/etc/init.d/pwrdd start-node http://hub:4444/grid/register

# ver el estado
/etc/init.d/pwrdd status-node

# detener
/etc/init.d/pwrdd stop-node
```

Configuración del servidor de Pandora

Independientemente del modo elegido, una vez iniciado, podremos empezar a asignarle ejecuciones de nuestras sesiones de navegación, agregando los parámetros de configuración de WUXServer al fichero de configuración de nuestro servidor de Pandora.

Suponiendo que hemos desplegado nuestro servidor PWRD en el mismo servidor en que corre nuestro servidor Pandora FMS, agregaríamos la siguiente configuración:

```
# agregar a /etc/pandora/pandora_server.conf
wux_host 127.0.0.1
wux_port 4444
```

Despliegue en Windows

Pasos a seguir para preparar un entorno de trabajo para Pandora WUX en Windows.

Nota: Recomendamos encarecidamente Linux para alojar el servicio PWRD.

Puede obtener los archivos necesarios para su despliegue de la librería de módulos en https://library.pandorafms.com/index.php?sec=Library&sec2=repository&lng=en&action=view_PUI&id_PUI=816

Despliegue de PWRD en sistemas Windows

Prerrequisitos:

- Descargar [PWRD server](#)
- Instalar Java 1.8 en la máquina que dará el servicio.
- Instalar Firefox 47.0.1 (descargable en <https://ftp.mozilla.org/pub/firefox/releases/47.0.1/>).
- Preparar un perfil de Firefox que se utilizará para probar las sesiones automatizadas (opcional): <http://toolsqa.com/selenium-webdriver/custom-firefox-profile/>
- Crearemos el directorio C:\PWR

Haremos la siguiente distribución de ficheros:

- Descompresión del fichero PWR_Server.zip a C:\PWR\
- Exportar el perfil de Firefox a C:\PWR\profile

El uso de un perfil para realizar los chequeos no es obligatorio, no obstante se recomienda su uso sobre todo en caso de utilizar proxy o querer emplear autocompletado de contraseñas. Para crear un perfil de Firefox seguiremos los pasos de las siguientes capturas:

Elección del directorio destino:

Debemos iniciar Firefox con el nuevo perfil para personalizar opciones como el proxy, mostrado de popups, etc:

A continuación instalaremos el servicio ejecutando el fichero BAT *service_installer.bat* proporcionado. E iniciaremos el servicio:

```
net start PWRSRV
```

Desde este momento ya tenemos el servidor de Selenium corriendo en nuestra máquina. No obstante podemos ejecutarlo a mano (parando el servicio previamente) en caso de que queramos realizar tareas de *debugging*, mediante el siguiente comando:

```
java -jar selenium-server-standalone-2.53.1.jar -firefoxProfileTemplate C:\selenium\firefox_profile -port 4444 -v
```


Grabar una sesión PWR

Antes de monitorizar una experiencia de usuario debemos grabar las acciones que queremos que el sistema realice, para navegar por nuestro portal web.

Para ello tenemos disponible la utilidad "PWR recorder" en la [librería de módulos](#).

Ejecute el archivo descargado para iniciar el entorno de grabación. Una vez iniciado el entorno, podrá acceder al grabador pinchando en el botón siguiente:

Deberá aparecer una ventana como la que aparece en la imagen:

Desde este momento podremos navegar por el sitio web que queremos monitorizar y las diferentes acciones de cada paso que avancemos irán apareciendo en el grabador.

Para detener la grabación utilizaremos el siguiente botón, situado en la parte superior derecha del grabador:

Una vez completadas las acciones, podemos realizar comprobaciones sobre la página, por ejemplo verificar la existencia de un texto determinado para asegurarnos de que la página cargada es la correcta. Para ello haremos click derecho sobre una sección de texto en la ventana del navegador mientras continuamos grabando, y seleccionamos la opción *verifyText*:

Aparecerá un nuevo paso en el grabador indicando la acción de comprobación de texto indicada:

Podemos reproducir la secuencia completa mediante el botón *Play entire test suite* y comprobar que finaliza correctamente:

Una vez verificada la validez de la secuencia de navegación, la guardaremos (Archivo -> Save Test Case). Este será el archivo que carguemos cuando creamos un nuevo módulo de análisis web en nuestra consola de Pandora.

Grabar una sesión transaccional para Pandora WUX

Pandora WUX permite dividir la monitorización de la navegación de un sitio web en múltiples módulos, que representarán cada uno de los pasos realizados.

Para insertar un nuevo punto de control y generar los módulos de fase (hasta ese punto) haga clic derecho en el punto donde desea identificar el comienzo de fase.

Como comentario pondremos el siguiente texto:

```
phase_start:nombre_de_fase
```

La fase englobará el tiempo y resultado de todos los comandos que se encuentren hasta el siguiente comentario:

```
phase_end:nombre_de_fase
```

Todos los comandos que se ejecuten entre una etiqueta `phase_start` y `phase_end` se englobarán dentro de esa fase.

Crear un módulo de análisis web en Pandora FMS Console

Una vez tenemos nuestra sesión de navegación grabada, es hora de desplegarlo como módulo en la consola de Pandora FMS.

Para ello accederemos a **cualquier agente asociado a un servidor con la funcionalidad WUX habilitada** y crearemos un módulo de tipo "Análisis web":

Una vez seleccionado "crear" deberemos rellenar todos los campos que nos presenta el formulario:

- Nombre del módulo: Módulo WUX (Padre), todos los sub-módulos con los resultados de la monitorización serán dependientes de este nuevo módulo.
- Sitio web objetivo (target web site): Para la monitorización de estadísticas de rendimiento de sitio web.
- Lanzar pruebas de rendimiento (Run performance tests): Indica que queremos monitorizar, no sólo la experiencia de navegación, sino también las estadísticas de rendimiento del acceso al sitio web objetivo.
- Histórico de datos: Guardar o no la información histórica de los módulos que monitorizan esta experiencia de usuario.
- Sección *textarea*: En este cuadro de texto deberemos copiar (o cargar) el contenido de los ficheros de la sesión de navegación que hemos grabado previamente.

Macros personalizadas

El funcionamiento de las macros personalizadas implica la substitución de determinadas cadenas de texto presentes en nuestro fichero de sesión de navegación, por ciertos valores personalizables.

En el caso de módulos de análisis web, se ha mejorado esta funcionalidad, permitiendo **macros dinámicas** que permiten que estos valores se traduzcan en **fechas y horas** variables.

¿Por qué esta funcionalidad?

Supongamos que necesitamos monitorizar, mediante una sesión de navegación, el correcto funcionamiento de una aplicación web de reserva de salas de reuniones.

Si establecieramos una fecha y hora fija a la hora de rellenar los datos del formulario, es posible que el sistema cancele la reserva porque en algún momento quede en el pasado, por ejemplo podríamos estar intentando reservar una sala para la semana pasada.

También es posible que nos encontremos con un máximo de tiempo para poder efectuar esa reserva, y que el sistema nos obligue a reservar la sala dentro de un período de tiempo concreto, por ejemplo, no superior al mes vigente.

Para evitar tener que editar la macro cada pocos días, y despreocuparnos del apartado de configuración, podemos usar las macros dinámicas, diciendo al sistema que reserve la sala siempre para el día siguiente al momento de realizar el test.

Para poder usar esta funcionalidad, los valores deberán tener un formato específico, logrando las siguientes sustituciones posibles:

```
@DATE_FORMAT (fecha/hora actual con formato definido por el usuario)
@DATE_FORMAT_nh (horas)
@DATE_FORMAT_nm (minutos)
@DATE_FORMAT_nd (días)
@DATE_FORMAT_ns (segundos)
@DATE_FORMAT_nM (mes)
@DATE_FORMAT_nY (años)
```

Donde "n" puede ser un numero sin signo (positivo) o negativo.

Y FORMAT sigue el standard de [strftime de perl](#)

Ejemplos:

```
@DATE_%Y-%m-%d %H:%M:%S
@DATE_%H:%M:%S_300s
@DATE_%H:%M:%S_-1h
```

Visualización de los datos

La información generada por WUX se visualizará en forma de módulos de la siguiente forma. Habilitar la vista en *modo jerárquico*, en la lista de módulos, nos ayudará a visualizar la información de forma mucho más clara:

Dentro de esta jerarquía encontramos los siguientes módulos:

- *módulo_Global_Status*: Indicará el estado **global** de la navegación completa.
- *módulo_Global_Time*: Indicará el tiempo **global** empleado en la navegación completa.
- *módulo_Global_Screenshot*: Contiene una imagen con el resultado del error de navegación, **solo se generará en caso de error**
- *módulo_Phase X: Nombre de fase_Status*: Indicará el estado de la navegación durante la fase X
- *módulo_Phase X: Nombre de fase_Time*: Indicará el tiempo empleado en la fase X

Ejemplo: Vista de captura de error.

Atención: Si ha actualizado desde versiones anteriores a Pandora FMS 7.0NG 712 debe realizar un pequeño cambio.

Para poder almacenar las capturas de pantalla generadas por WUXServer, deberá lanzar las siguientes operaciones en su esquema de base de datos:

```
alter table tagente_estado modify column datos mediumtext;  
alter table tagente_datos_string modify column datos mediumtext;
```

Si no realiza estas acciones las capturas podrían verse cortadas.

Accediendo a la sección *WUX* del agente podemos ver detalles adicionales de la transacción y sus fases:

En la captura anterior puede verse la diferencia en tiempos de las dos fases en que se divide nuestra grabación, así como las estadísticas de acceso al portal frontal.

Las estadísticas del sitio web se resumen en los siguientes conceptos:

- **Stats_TT**: Tiempo total en obtener el sitio web.
- **Stats_TDNS**: Tiempo total en resolver la dirección IP del objetivo.
- **Stats_TTCP**: Tiempo empleado en conectar vía TCP.
- **Stats_TSSL**: Tiempo empleado en establecer comunicación SSL.
- **Stats_TST** : Tiempo transcurrido hasta que inició la transferencia de datos.
- **Stats_TTC** : Tiempo transfiriendo datos, agrupará todos los tiempos de transferencia de recursos.
- **Stats_TTR** : Tiempo empleado en transferir el contenido de la página.
- **Stats_TTR_main**: Tiempo empleado en transferir el código HTML.
- **Stats_TTR_image**: Tiempo empleado en transferir los recursos de tipo imagen (png|jpg|jpeg|bmp|tiff|gif|webp|svg).
- **Stats_TTR_css**: Tiempo empleado en transferir las hojas de estilo.
- **Stats_TTR_js**: Tiempo empleado en transferir los ficheros JavaScript.

Asignar alertas a un módulo de análisis web

Las alertas asociadas a los módulos de análisis Web siguen la misma dinámica de funcionamiento que todo el sistema de alertas en Pandora FMS.

Por compatibilidad, se recomienda asignar las plantillas de alertas sobre los sub-elementos autogenerados por el módulo de análisis web, como por ejemplo:

- El estado de la navegación global
- Alertas sobre umbrales de tiempos
- Alertas con plantilla de aviso puesta en "siempre" para los módulos captura de imagen de resultados

Nuevos widgets para Dashboards

Para poder representar la información de la navegación de usuario, se han agregado dos nuevos tipos de widget:

Widget de estado de las fases de navegación

Widget de estadísticas del sitio web

