Hands-on chaos testing with OpenText Professional Performance Engineering

Take advantage of OpenText Professional Performance Engineering for all of your chaos engineering needs. Read now for a step-by-step guide on how to get started with this integration.



Contents

Getting started with OpenText Professional Performance Engineering	3
Adding chaos testing	4
Creating a Gremlin scenario for chaos testing	6
Performing the chaos test using controller	9
Test results	12
Turning chaos into confidence	13

A distributed services approach entails countless variables that even the most meticulous development team cannot predict. Even when each service functions perfectly in a vacuum, interactions between them can be—in a word—chaotic. So, to most effectively build infrastructure resilience against the unpredictable, test engineers developed <u>chaos engineering</u>.

This disciplined form of software engineering prioritizes the knowledge that failures are inevitable and that systems can be designed to withstand them without significantly affecting the end-user experience—even at scale. A primary component of this process is chaos testing, which simulates failures in a system so that developers can build resilience for the broadest possible array of failure scenarios.

To further explore this concept, this guide will walk through using Gremlin to add some chaos to an application and <u>OpenText[™] Professional Performance</u> Engineering to analyze the application's ability to handle the failures.

Getting started with OpenText Professional Performance Engineering

To follow this tutorial, ensure that you have the following prerequisites. Additionally, note that these services are currently compatible with Windows only.

- OpenText Professional Performance Engineering (free trial available)
- Gremlin
- The OpenText[™] VuGen tool for simulating typical user behavior on an application
- A web application to test. This tutorial uses this <u>Spring Boot application</u>, but you can use any app to which you have the rights.

Before proceeding, follow <u>these steps</u> to install and configure Gremlin on your machine.



Adding chaos testing

Setting environment for chaos testing

To get started, you'll need to enable OpenText Professional Performance Engineering to simulate virtual users using the OpenText Professional Performance Engineering VuGen (Virtual User Generator) script.

Open the VuGen App on your Windows system. Click the + symbol at the upper-left corner and select the Add New Script option.

* • + • Solution Explorer • # # ?	< Start Page ×			
	RECENT ACTIVITY	WHAT'S NEW	LEARN MO	RE
	Chaostest 5/9/2022 Local	LoadRunner Professional & Protocols	What's new in the 2022 release - Windows 11 support - New Kaka precoci (replay only) - Choos fording with Greenin events - Jana 17 support - Criticx Workspace 2109 support (replay only)	LoadRunner Beveloper & Devvlob >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
		ക	C 1/2 > READ MORE	చి
	& හි හි in f €.	LoadRunner	What's new in the 2022 release	LoadRunner Professional video
	Output	Options *	н. Н	* 3

From the **Create a New Script** window, select the **Single Protocol** option under the **Category**: tab at the far left.

Then, under Protocol:, select the Web - HTTP/HTML option.

Next, provide the name and location of your script. Your screen should look similar to this:

Category:	Protocol:		Filter:	E 8=
✓ VuGen	3C		SAP - Web	💏 TruClient - Native Mobile
Single Pro	cle - 2 Tie	PL	🔽 SAP GUI	TruClient - Web
Multiple F	rotocols d IoT		😪 Siebel - Web	Tuxedo
Popular	:le NCA		SMP (SAP Mobile Platform)	S Web - HTTP/HTML
Recent	'3 (Post (Office Protocol)	SMTP (Simple Mail Protocol)	🗞 Web Services
	(Remote	e Desktop Protocol)	PCoIP Teradici PCoIP	
	(Remote	e Terminal Emulator)	TruClient - Mobile Web	
Script Name:	WebHttpHtml1			
ocation:	E:\Winter 2022\Ex	tra\Chaos Testing		
Solution Name:	Enter a solution na	ame or leave blank to c	reate an untitled solution	Create folder for solution
Solution Target:				

Once you've filled in the required fields, click the **Create** button.

If the application doesn't do so automatically, use the **Solutions Explorer** pane to navigate to **Chaos Test > Actions**. Then, double-click on **Action** to add the code below. Note that the specified URL is for the localhost, as this is where the demo web application is hosted.

```
Action()
{
Ir_start_transaction("01_HomePage");
// Replace the URL with your server IP address
web_custom_request("web_custom_request",
"URL=http://127.0.0.1:8080/",
"Method=GET",
"TargetFrame=",
"Resource=0",
"Referer=",
"Body=",
LAST);
```

Ir_end_transaction("01_HomePage", LR_AUTO);
return 0;
}

Your VuGen window should now look similar to this:

File	Edit	View	Search	Desi	3u	Record	Replay	Tools	Integration	is Windo	w He	elp								7	8	×
*-	+~ 8	1~	6 A	ា	Q	•	0	a	Þ 20	esign Studio	12	Ø	QI									
Solution	Explorer		* 0 ×	Chao	e Taet ·	Action c	×															
	tion Untitled Notare Test Web & Actions & waver, index & waver, and & waver, and & States Flees Deata Flees Deata Flees Parameters Recording Recording Recording Recording Recording Record Recording Record	ib - HTTF	PHTML]	2 1 2 3 3 4 5 6 7 7 8 9 9 10 11 12 13 14 15 16 17 17 17 17 17 17 18 19 19 19 10 11 11 12 13 14 15 16 10 10 10 10 10 10 10 10 10 10	- Act	ion() Ir/// /// web_cus 'Me 'Ta 'Re 'Bo LAS Ir_end_ return	<pre>start_tran Replace th tom_reques</pre>	saction("4 e URL with t("web_cur t("web_cur z.e	03_HomePage") h yoar server ston_request" £8882", ¢Page", LR_AU	IP address												
				Output																		а×
				Replay		~			o Q Search		Op	tions +	64									
				E Repl Linuting Vuser	ay status neccio Termin	passed \	ñew summar enu .	Started at:	5/9/2022 4:59:42	PM Elapsed tin	ne: 00:00:0	02										
Solution	Explorer S	tep Navig	gator	Output	Task	Errors	Runtime	Data														
Ready														S Replay	status passec	Script part	ing succee	fed	ln 17	col 1		

The OpenText Professional Performance Engineering script above simulates a user-sent GET request to the specified URL.

Click the \triangleright (Run) button on the toolbar to check whether the script runs without errors. If it runs successfully, VuGen will open a Summary tab similar to that in the following image. Otherwise, it will prompt you to resolve any discovered errors.

File Edit View Search	Design Record Replay Tools Integrations Window Help	- ø ×
*- +- 🗈- +	• 🗗 😼 💿 🚺 み Þ 🖾 Désign Studio 📜 🔯 🍳	
Solution Explorer • # X	ChaosTest : Replay Summary × ChaosTest : Action.c	8
	Script Name: Chaos Test Sorge passed Expeed time: 00 00 00 728 hh mm ss ms Started at: 580/002 5 18 37 PM Ended at: 580/002 5 18 37 PM	Save MAR File Export to PDF
	For additional details, enable the collection of replay statistics. Choose: Tools > Cystons > Scripting > Replay (solect "Collect replay statistics")	
	These statistics include transaction breaktionn information, connections, and various HTTP-related items. New E, Soalburg the option may affect the register performance. For additional network traffic details, enable the generation of W insights Report. Choose	
	Output	- 0 ×
	Replay V X 🖂 V Locate Q. Search term 0 0 O Options V 🖬	
	Replay status passed Vew summary Stanted at 59/2022 5:18:37 PM Elapsed time: 00:00 00 Uniting Section 1995Emil. Vuser Terminated.	
Solution Explorer Step Navigator	Output Tasks Errors Runtime Data	
		Replay status passed Script parsing succeeded In 0 col 0

You've now successfully created a script that enables OpenText Professional Performance Engineering to simulate virtual users.

Creating a Gremlin scenario for chaos testing

Next, for this demonstration, you'll create two attacks:

- A CPU attack to assess how the hosted web application performs when it uses nearly all available CPU resources.
- A latency attack to test the application's responsiveness over varying network conditions.

Once you've finished signing up for Gremlin, you need to create a team. This will help you access the Gremlin scenario using its Team ID and API key.

Note that you can view the installation <u>documentation</u> to ensure that you've properly configured the application for Windows.

Now, click on **Dashboard** from the left-hand sidebar. Then, click the **Create Scenario** button to begin creating a chaos testing scenario.

ad)Grendin X 🛨								
i app.gremlin.com/dashboard								1 🖬 🔁 Guest 🕴
Dashboard							•	Halt All Attacks
Technology Stack Reading	ess							
Get started by addre Explore each component to te	ssing common failure	modes using Gremlin's I	Recommended Sce	narios.				
Containers Orchestration	Cloud Compute	Monitoring	Messaging	Databases				
\bigcirc	R	ß	iii					
Never attacked	Never attacked	Never attacked	Never attacked	Never attacked				
Link your atta	cks together	Not sure v	where to start?		Quickly run an attack			
Create S	icenario	View Recomm	nended Scenarios		Create Attack			
Services			Targets		Previsioned SOMey	VEWALL +		
Automatically dis	cover your services		ACTIVE fe80:75	0e:8723:1821:706b		Attack		
Uncover the unknowns and wea	ak links in your environment. Spe	nd less time matching VMs						
Enable services discovery								
Total Attacks Run		THIS WEEK *	Recent Scenarios			,		Q
	(control of the second	for the service of t	from a final definition of the second definiti	Image: Control of the second seco	Image: Control of the second seco	Image: Control of the state with the sta	Image: Control of Cont	Contraction Contraction

If you've configured your Gremlin app and teams correctly, you should be able to see the number of hosts connected to the server. The image below shows that one host is available to test the scenario on the Windows system:

Sce	narios > New				Halt All Attacks
BB Dankboard	Add a new attack Define a custom attack	Add a recent attack Recently completed attacks	Add a Status C Check system rea	Check	
© Services Services	What do you want to attack?				
Canado ana	Hosts Invaliable	Containers Devailable	W Kubernetes		
Afraeda	Choose Hosts to target Specify the coverage and details for impact				
Schedules E Clients	Q. Narrow the number of potential largets by Target all hosts	Expend Al	BLAST RADIU 1 of 1 HOSTS TARGETED	US () HOSTS IMPACTED	
© Reports	Operating Systems Rindows	0			
Bees \$ xx ↓	2 - Course Lagre				\$

Click Add a new attack to open the Choose a Gremlin section.

Then, select the Resource radio button from the Category list and click CPU from the Attacks list.

Next, set the Length of the attack to 120 seconds and ensure that the CPU Capacity value is set to 100.

Now, select All Cores from the dropdown menu to ensure all available cores are used for testing.

s 🐨	cenarios > New	Hait All Attacks
88 Daahbaard		
© terrices	Cocces a Gamilia fonci hi typer inici s v visua.	
Scenarios	Category Annaba	
Garrie Days	Resource Test sparse faulder of larges in consumption of computing resources. ()	
17 Attacka	Data Deal Instagated unexpected changes in your environment such as power outlages, nob falleres, doi: 40t, or sestilation chankes. Data Data Instagated unexpected changes in your environment such as power outlages, unexpected unexpected outlages, and unexpected outlages, an	
C. Schedules	Test sparet unveilable retwork constitutes.	
() Clienta	Minuty Type of the strength of the st	
© Reports	Langh The import of the attack desired 120	
 Docs	CPU Celastity The permitting of CPU to consider or investigation The permitting of CPU to consider or investment and	
¢\$ MR	At Cores	
6	Consume at DRUgoes	
	And the County Callons	•

Finally, click the Add to Scenario button.

Your **Dashboard** should now appear similar to what's shown below:

Scena	arios > New					Balt All Attacks
	Chaos Test					
	Description CPU and Latency attack					
•	CPU 2 minutes	100% off	All Cores: Yes	CPU Capacity: 100		
(*)	∲ [⊙] Add a new attack Define a custom attack	4 Add a recent attack Recently completed attacks		O Add a Status Check Check system readness		
Serve	Scenario			Grander API Exam	-pies	

You can name your scenario and add a brief description, but don't save the scenario just yet—you'll now add the latency attack.

Once again, click Add a new attack.

This time, select **Network** from the **Category** list and click **Latency** in the **Attacks** section.

Then, update the Length of the attack to 120 seconds and keep the delay at 100 **MS** (milliseconds). You can leave the rest of the fields blank or add optional details for your specific scenario.

s > New		00
State Test against unexpected changes in your environmen node failures, clock onit, or apprication crashes. Network Test against unellable network conditions.	sch a gover nutgen. Of Star 2012 All the set of subsch bit halse 37 (Sarvers et 2013) Of Star and provide set of subsch bit halse 37 (Sarvers et 2013) Of Star and St	
	Polist Lots (maximum) Add no system tands are experience when a periodicipal of subcond velocity polishet are tropped or considered	
Length The length of the attack (seconds)	120	
MS How long to delay egress packets (millis)	100	
IP Addresses Only inpact mills to these IP addresses folder shaft har reported and and the control comp. (Later carries,		
Device Impact truths over this national interface books to fee and theods: and 1		
Hostnames Only impact by this to these had names factors when the impact of a codes," Also gave an other (1) also gaving		
Remote Parts Only impact adapting traffic to these semale parts large server within (this)	*63	
Local Ports Only impact outgoing traffic from these local ports Areas writing failed date		
Providers Educated service providers to affect	Service Providera 💌	
Tags Only impact traffic to hosts running therein clients associated with these tags	Host Tage .	
	SHOW ADVANCED OPTIONS >	

Once finished, click the Add to Scenario button.

Your window should now look like the image below:



Finally, click the **Save Scenario** button. Now, when you click on the **Scenarios** button from the sidebar, you should see a section named for your created test scenario.

Performing the chaos test using controller

Now you're ready to execute your chaos test using Controller.

When you open Controller on your Windows machine, the program will open a **New Scenario** window. Select **Manual Scenario** under **Scenario Type**.

Next, add the VuGen script created earlier. Click **+ Add Scripts** and navigate to the folder where the script is saved. Select the script file and click the **ADD** button.

New Scenario Scenario Type		© ×	
Manual Scenario Manage your load test by specifying the number of virtual users to run.	Goal Oriented Scenario Allow LoadRunner Controller to create a scenario based on the goals you specify.		
O By Number By Percentage + Add Scripts Record Script ALM	- 	(Scripts: 1)	
Script Name	Script Type		
ChaosTest	Multi Protocol		

Once you've added the script, Controller will open it, as shown in the image below:

Scenario Scripts			Service Level Agreement
Sicript Name	Script Path 2 Winner 2022 Exter(Chaos Textro)(Chaos Text	% Load Generators 100 % AllLoad Generators	Mere Control Flat X (new Cardinal Cardinal Control (Cardinal
Scenario Schedule		10maae	
Schabe Inere Control C	Sule C Basic schedule bottos a nues NASIS MASIS DO DO DO (PELMA SS)		00.02.00 00.00.00 00.04.00 00.05.00 00.06.00

You'll now need to define a few criteria in the **Global Schedule** panel at the lower-left corner of the window.

First, double-click on the **Start Vusers** section and set the **Start** value to 50 Vusers. This is the maximum value that OpenText Professional Performance Engineering Community Edition permits.

Additionally, set the **(HH: MM: SS)** parameter to 00:00:01. This represents one user logging in per second to maximize the chaos introduced into the system.

Then, click **Next** to move to the **Duration** field. Set it to 00:04:00. This means that the test will run for four minutes.

Next, navigate to the **StopVusers** field and select the **Simultaneously** radio button. This will create an event in which all users exit the app at the same time.

Finally, click the **OK** button.

You should now notice that the graph indicating Vusers versus Time has changed:

LoadRunner Controller - Scenario1 -	(Design(1))							- 0	
e View Scenario Results Diag	nostics Tools Help								
📽 🖬 🏦 😵 🔥 🏯									
Scenario Scripts					Service Level Agreement				
N 19 19 19 19 19 19 19 19 19 19 19 19 19						New BDetais /Eds XDelete PAdvanced			
Script Name Script Path		*	% Load Generators			Senire Level Amagnet			
G Micheostest	Eykine 1020Ener(Chos Testrg(ChosTest	100 %	cAll Load Generators>	=	Servoir over all agreement Counterfly in GE Auroland and east Color the New bottom to define SLA othere for your lead text				
enario Schedule X R2 G G redule Name: Schedule 1 sodule by: @ Scenario n Mode: @ Real-world sc	hedule C Basic schedule	× 60 50						Legend II — Global Schedule	
/bol Schedule			/						
🥐 🧠 🔶 👘 Total: 50 Vusers		40	/					- 1	
Action Properties									
Start Vusers Start 50 Vusers 1 or	erc (00.00.01 (HH MM SS)	§ 30	1						
Duration Run for 00.04.00 (HP	(MM SS)		/					i	
Stop Vusers Stop all Vusers smult	teneously	20							
		10							
		00:00:0	00:01:00	00:0	02:00	00 00:03:00 Time	00:04:00	00:05:00	
Design	Run						Ra Auto Collete Res	uto	

Note that since you're testing for chaos, you can configure the test to ignore think time. To do so, locate the checkbox for your script under the **Scenario Scripts** section in the upper-left corner of the window. Right-click the white space near the checkbox and navigate to **Runtime Settings**. Then, in the **General** section of the new pane, click on **Think Time**. Select the radio button for **Ignore think time**.

To add the chaos scenario created in the Gremlin app, click the rightmost button directly beneath the **Scenario Schedule** header. This is the **Disruption Events** button.

S	cenario Sche	dule				
	'×∞3	Z				
Sc	hedule Name:	Schedule 1 Disruption Events		•		
Sc	hedule by:	Scenario				
Ru	In Mode:	Real-world schedule	C Basic schedule			
Gl	obal Schedule					
1	🤣 🎭 🕈 🔸	Total: 50 Vusers				
	Action	Properties				
	Initialize	Initialize each Vuser just before it runs				
	Start Vusers	Start 50 Vusers: 1 every 00:00:01 (HH:MM:SS)				
	Duration	Run for 00:04:00 (HH:MM:SS)				
▶	Stop Vusers	Stop all Vusers simultaneously				

Then, click the **+AddEvent** button and provide the API key and Team ID of the team you created in your Gremlin account.

Once you add the Team ID and API key, you'll see the created scenario as shown in the image below.

Select the event and click **ADD** button to add it to the controller scenario. Set the **Start Time** to 00:01:00. This means that the disruption event will start one minute into the scenario.

Add Event					@ X
Type All					
API Key 24d3ab98e7f0eff5	9152db36689l	of7b6594e97bae81e0d03b1486	of Team ID 0340ed-	36cc-4a55-8340-ed36cc0a551d	
Technology type	-	Attack Type	- Q		
Event Name			Туре	Created by	
🗹 Chaos lest		c	pu/latency		
				ADD CA	ANCEL

Next, click the floppy disk/save icon below the top menubar and provide a name for your saved scenario. The demonstration has kept the name as Scenario1.lrs.

Select your destination folder for the .lrs file and click Save.

Now, you can run the scenario by clicking the **Start Scenario** button toward the upper-righthand corner of the window.

OpenText Professional Performance Engineering will then produce graphs of the results and display them as shown below:



Test results

Once the scenario has finished running, click the **Reports** option from the menubar and select **Analyze Results**.

Once all the summarized results have been generated, you'll receive a pop-up to view them.

Click Yes. This will open graphs displaying Hits per Second, Throughput, Average Transaction Response Time, and several other statistics.

🙃 LoadRunner Analysis - Session1.Ira			- 0 X
File Edit View Graph Reports To	nols Windows Help		
Session Explorer 4 X	St. C. 1 2 C. 1 S. C. 1 S. C. 1 S. C. 1 2 C.		4.6
Bit X Image: Second state Image: Second state Image: Second state	09/05/2022 21 15 04 0 4m 523 Datation Scenario1	E Fes Result Name	User Notes 🔠 Raw D
- (+, Transaction Summary - (+, Average Transaction Res)	SCENARIO SCHEDULE 60	STATISTICS	an a
	40	Maximum Running Vusers Total Throughput (bytes)	ph Data
Properties # ×	20	Average Throughout (B/s) Total Hits	
Description Summary page sh Filter (Include Think Tim Percentile 90		Average Hits per Second Passed Transactions Ratio	
ine Summary Report	TRANSACTIONS		
Summary page shows the summarized analysis on a single page			
Controller Output Messages			

You can now compare the output data.

Click on the Throughput graph, then right-click on the chart.

Then, select the **Merge Graphs** option and choose the **Hits per Second** graph from the dropdown. Click **OK** to see the detailed diagram as shown in the image below.



The chart indicates that for the first minute, the application was up and running with high throughput and hits. Then, after the chaos started, it dipped sharply around 01:28, seesawing dramatically over the next 90 seconds, at which point the chaos event ends.



Merging **Throughput** and **Average Response Time** shows an inverse relationship between throughput and response time because of the chaos:

Your results will likely vary depending on your system configurations and scenario values. Still, the detailed analysis of how well the configured app handles the chaos is invaluable in a production environment. It can enable you to determine what triggers which failures and develop solutions to manage similar real-world events.

Turning chaos into confidence

Chaos engineering is crucial for preventing the breakdowns of services that can eventually lead to full application failures. Now that you've experienced some of the capabilities that chaos testing can provide, you can more clearly identify ways to integrate it into your development and testing cycles. As organizations continue to shift to distributed architectures, the benefits of this engineering, testing, and analysis make the practice a principal component in building resilience within chaotic systems.

Learn more at

OpenText Professional Performance Engineering

OpenText DevOps Cloud

