SandStorm AppDynamics Integration

Getting Started Guide

Contents

Preface	.3
Application Performance Testing using SandStorm	.3
Step 1: Register with SandStorm Cloud Application	.3
Step 2:Provide AppDynamics details to sandstorm	.4
Step 3: Configure naming rules	.4

Preface

This document is designed to help you getting started on using SandStorm (AppDynamics plugin) for identifying performance bottlenecks during performance test execution. This plug-in allows users to drill down into performance test results and isolate code bottlenecks that impacts performance. It lists the steps to configure plugin in SandStorm, configuration changes in and result analysis. Should you have any queries, please write to us at sandStorm@impetus.com.pacts

Application Performance Diagnostics using SandStorm AppDynamics plugin

SandStorm AppDynamics integration is used to integrate the performance test execution and diagnostics. It allows user to identify code performance bottlenecks from the performance test results. You need to have the following pre-requisite to use the plug-in.

- a) Valid account for SandStorm Cloud application
- b) Valid account for AppDynamics SAAS offering or local AppDynamics Pro installation

The steps for analyzing diagnostics information during performance test execution using the plugin are given below.

Step 1) Provide AppDynamics details to SandStorm

<u>Step 2)</u> Download naming appDynamicsConfiguration.xml and import in AppDynamics. If applications are already created in AppDynamics, manual add the naming rules to identify SandStorm transactions

Step 3) Execute scenario and result analysis

Pre-requisite: Register with SandStorm Cloud Application

Register with SandStorm Cloud Application by accessing the URL mentioned below. http://sandstorm.impetus.com/sandstorm_cloud_version. The SandStorm Evaluation page appears. Enter the mandatory details. If you are a new user, then select the 'New User' option and enter the required details, and submit the registration form. Once you are registered user with SandStorm Cloud Application, you can login into the application using the credential shared with you over a mail. Please navigate to SandStorm Cloud Application using the given below https://sandstorm.impetus.co.in/sandstorm/login

To execute any performance test using SandStorm Cloud Application, you need to create a project and a scenario inside that project.

Once you have the pre-requisites set up you can configure SandStorm AppDynamics integration.

Step 1: Provide AppDynamics details to SandStorm

After successful registration with SandStorm Cloud Application, provide the following information related to AppDynamics in SandStorm Account Settings:

- 1) AppDynamics Controller URL
- 2) AppDynamics username. (On-Premise AppDynamics users should append @customer1 at the end e.g. if username is sandstorm then provide sandstorm@customer1).
- 3) AppDynamics password.
- 4) AppDynamics application (Application in which diagnostic data is stored).

Account Settings

Γ	Integr	ration								
	AppD	ynamics								
									6	
		User Name *	sandstorm@customer1							
		Password *	•••••							
		Protocol *	HTTP •							
		URL*	impetus-1286							
		Port No *	8090							
		Application Name *	travelPortal							
						Test C	onnecti	on	Save	

Screen 1: AppDynamics configuration in SandStorm

Step 2: Configure naming rules

Download the sandstorm-appdynamics configuration file from downloads.



Screen 2: AppDynamics SandStorm Business Transaction Rules XML download

Once the appDynamicsConfiguration.xml is downloaded successfully, import the application in appdynamics using this xml file.

Based on the AppDynamics applications, you need to import the naming rules to detect SandStorm transactions. The following two cases might arise:

- a) Applications do not exist in AppDynamics
- b) Applications exists in AppDynamics

If you are creating a new application, then follow the below steps to configure naming rules:

1) Download the rules xml file from SandStorm Download section.

Screen 2: Download naming rules XML

2) Login to AppDynamics controller

Firefox 🔻	🔍 AppDynamics	+		Sec. 1	-		p langtonic in Coupling Num	Manual Sold
🗲 🛼 impe	tus-1286:8090/controller/#/location=APPS_ALL	DASH	IBOARD					☆ ⊽ C
							🔍 AooDunamics	
						Hearnom		
						Passwor	d sandstorm	
						1 455 W01	Remember me on this computer	
							Login	
							Forgot your password?	

Screen 3: AppDynamics Controller

3) Click on Import Application



Screen 4: Import Application in AppDynamics

4) Browse and select the rules XML downloaded from SandStorm and provide a name to the application

Application Import		×
Please select an App	plication XML file that was exported using the export tool.	
File name	blocklist.xml Select Application XML File	
File size	10326 bytes	
Application Name	Demo	
	Cancel Import	
Analyze Conligue	ACTIONS A	_

Screen 5: Import Application Dialog

- 5) Click "Import".
- 6) Reset the agents to send the data to this application.

If you have an existing application, then follow the below steps to configure naming rules:

1) Login to AppDynamics

controller

Firefox 🔻 📮	AppDynamics	+	aller agine		Satisfactoria da Tangataliy Malif	Manual Inc.
🗧 🛼 impetus-	1286:8090/controller/#/location=APPS_AL	_DASHBOARD				☆▼
					🔒 AppDynamics	
				Username	sandstorm	
				Password	•••••	
					Remember me on this computer	
					Login	
					Forgot your password?	

Screen 6: AppDynamics Controller

2) Open the required Application by clicking on the Application name

impetus-1286:8090/contr	roller/#/le	ocatio	n=APPS	_ALL_D	ASHBOARD&timeRange=last_15	_min
🕺 AppDynamics Pro						
Applications	•	◀		G	sandstorm	
МуАрр	3	D	aabbaar	d	Top Rusiness Transactions	Т
sandstorm	1		asriboan	u	Top Dusiness Transactions	
shoppingCart	Applic	catio	n Flow	Map 🕻	•	
Screen 7: AppDynamics Application	_					

3) Click on Configure



Screen 8: Application Configuration

4) Click Instrumentation and select application

S. AppDynamics Pre									
MyApp -	🚖 🚽 📄 🗛 MyApp 🕨 Configu	re Instrumentation							
Business Transactions	Transaction Detection Back	and Detection Fin	d User Experience	Free Detection Data Collectors Call Grach Settings JMX Memory Monitoring					
➤ Servers	Relati Androdon or Tax								
Events	Customized?	MyApp							
Troubleshoot	v A Muleo								
> Alert & Respond	E condition Day			Java - Transaction Detection NET - Transaction Detection PHP - Transaction Detection					
Analyze	Landstorm0464								
Contigure	P and a state	copy	Contigure an Tie	rs to use this consultation					
Slow Transaction Thresholds		• Entry Points	Entry Points						
Baselines		Туре	Transaction Monitoring	Automatic Transaction Detection					
		Servict	Enabled	Oticover Transactions automatically for all Servict requests Configure Naming Enable Servict Filter Detection:					
		Struts Action	Enabled	Clocover Transactions automatically for all Shulla Action Invections Transactions will be named: ActionName MethodName					
		Web Service	Enabled	Chocker Transactions automatically for all Web Service requests Transactions will be named. ServiceName OperationName					
		POJO	Any Java method can be the entry point for a Business Transaction. The class to which the method belongs to can be picked using different parameters like its name, its annetations it has.						
		Spring Bean	Enabled	Ciscose Tanaactions automatically for all Spring Beam Investigions Transactions will be named. Boarshare Methodhame					
						au 1	Enabled	Clocover Transactions automatically far all EUB invocations Transactions will be named: EUBName Methoditane	
		JMS	Enabled	Clacoler Transactions automatically for all incoming JABS Messages Transactions will be named. Destination Name or Listener Class Name (If Destination Name net available.)					
		Binary Remoting	Enabled	Cliccover Transactions automatically for all Binary Renoting requests (Thrift) Transactions will be named. RenoteInterface/ClassName.methodPlane					

Screen 9: Configuration for Agent

5) Click on Configure Naming

Rules	
Servlet Transaction Naming Configuration	×
What part of the URI should be used in the Transaction Name?	3
Use the full URI	
 Use a part of the URI (for example, if you have dynamic URIs) 	
Use the first v 2 segments of the URI in Transaction Names What does this do?	
Name Transactions dynamically using part of the request	
Use URI segment(s) in Transaction names	
Segment Numbers Enter a comma separated list of parameter numbers (e.g. 1,3,4)	
Use a parameter value in Transaction names	
Parameter Name	
Use a header value in Transaction names	
Header Name SandStorm-Transactic	
Use a cookie value in Transaction names	
Cookie Name	
Use a session attribute value in Transaction names	
Session Attribute Key	
Use the request method (GET/POST/PUT) in Transaction names	
Use the request host Transaction in names	
Use the request originating address in Transaction names	
Apply a custom expression on HTTPServletRequest and use the result in Transaction Names Explain This	
Cancel Save	ð
Trunscisne mie versienen ander and	

Screen 10: Configure Naming Rules

- 6) Add the following rules
 - a. Use a header value in Transaction name: Header Name : SandStorm-Transaction

After completing the above steps, SandStorm will be able to fetch diagnostics data for performance test execution.

Note:

1. You need to reset all the agents on the appservers to send the data to the specified application. For more info about agent-appserver runtime configuration visit:

http://docs.appdynamics.com/display/PRO14S/Java+Server-Specific+Installation+Settings

2. If the transactions are more than 50 than you need to increase the no. of transaction in the agent configuration file (), else it will collect data to 'All other transactions'. For more info about agent configuration visit:

<u>http://docs.appdynamics.com/display/PRO14S/App+Agent+for+Java+Configuration+Properties#AppAgentforJavaConfigurationProperties-CreatingandRegisteringTiers</u>)

3. This manual is prepared keeping java-agent in consideration; however for php-agent except agent configuration every step is same as mentioned above.

Step 3: Execute Scenario and Analyze Results.

- 1. Design the scenario using SandStorm scenario tab
- 2. Execute the scenario.
- 3. After the scenario execution, generate Analysis for the scenario from the Results tab.
- 4. Click on the AppDynamics link under Diagnostics header in the result navigation side bar. You can view the details of app-servers in corresponding tab.

	Sandstorm Transaction	Avg Response Time (s) 🖨	Max (s)	Min(s)
+	travelPortal.payment	21.25	52.832	21.014
+	travelPortal.CancelBoooking	21.235	26.112	21.071
+	travelPortal.MyBookings	6.586	41.045	1.898
+	travelPortal.load	6.414	18.658	4.279
+	travelPortal.search	0.395	3.77	0.335
+	travelPortal.book	0.248	6.746	0.085
+	travelPortal.navigate	0.148	2.579	0.096
+	travelPortal.resgisterPage	0.121	2.849	0.082
+	travelPortal.searchSuccess	0.07	0.858	0.055
+	travelPortal.login	0.038	0.565	0.027
þ	φ	🛯 🛹 Page 1	of 2 🍺 🖬	

Screen 11: Diagnostic results in SandStorm

- 5. You can view the transaction results in corresponding tab.
- 6. Drill down the transaction by clicking the + button.

-	travelPortal.MyBookings	6.586	41.045		1.898
	Business Transaction	Tier	ART(ms)	Calls/min	Errors/mi
	/travelportal/managebooking.travelPortal.MyBookings	travelPortalDev	6241	1	0
Ľ	/travelportal/passenger.travelPortal.MyBookings	travelPortalDev	8	1	0
	φ				

Screen 12: Drill down results in SandStorm

7. To view the graph click on transaction name .



Screen 13: Performance analysis of slow business transaction

8. Click on External link to get the detailed diagnostic information. This action will launch the AppDynamics controller in a separate window for detailed performance debugging.



Screen 14: AppDynamics transaction flow map

		Time	Exe Time (ms)	URL	Business Transaction	Tier
		26/05/14 4:55:44 PM	148848	/travelportal/managebooking/mybooking	/travelportal/managebooking.travelPortal.MyBookings	travelPortalDev
0	ili _j ili	26/05/14 4:55:43 PM	801	/travelportal/	/travelportal/	travelPortalDev
0	:=	26/05/14 4:55:33 PM	2981	/travelportal/	/travelportal/	travelPortalDev
0	:=.	26/05/14 4:55:33 PM	8592	/travelportal/	/travelportal/	travelPortalDev
0	÷=.	26/05/14 4:55:30 PM	5720	/travelportal/	/travelportal/	travelPortalDev
×		26/05/14 4:55:26 PM	165538	/travelportal/managebooking/mybooking	/travelportal/managebooking.travelPortal.MyBookings	travelPortalDev
0	÷=.	26/05/14 4:55:21 PM	776	/travelportal/	/travelportal/	travelPortalDev
0	:=	26/05/14 4:55:17 PM	758	/travelportal/	/travelportal/	travelPortalDev
×	iliji	26/05/14 4:55:05 PM	21140	/travelportal/payment/process	/travelportal/payment.travelPortal.payment	travelPortalDev
×		26/05/14 4:54:20 PM	21064	/travelportal/payment/process	/travelportal/payment.travelPortal.payment	travelPortalDev
×		26/05/14 4:54:20 PM	21064	/travelportal/payment/process	/travelportal/payment.travelPortal.payment	travelPortalDev
×		26/05/14 4:54:16 PM	21126	/travelportal/payment/process	/travelportal/payment.travelPortal.payment	travelPortalDev
×		26/05/14 4:54:15 PM	21849	/travelportal/payment/process	/travelportal/payment.travelPortal.payment	travelPortalDev
×		26/05/14 4:53:17 PM	108	/travelportal/booking/booksubmit	/travelportal/booking.travelPortal.book	travelPortalDev
-			000			
				URLs	SandStor	m Transactions

Fig 15: AppDynamics drill down for a slow transaction

Execution Time: 21140 ms. Node travelPortalDev. Timestamp: 26/05/14 4:55:05 PM.

Set as Root F	Sho	w Filters 🔻	ρ		
	Time (ms)			
▼ 🛱 org.springframe	5 ms (self)	0 %			
TTPServle	et:service:641		0 ms (self)	0 %	
🔻 🦰 Servlet -	spring:doSer	vice:856	0 ms (self)	0 %	
🔻 🗖 Servi	let - spring:dol	Dispatch:925	0 ms (self)	0 %	
🔻 🌏 Sp	0 ms (self)	0 %			
▼ 🥝	Spring Bear	- org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter#0:invokeHandleMethod:745	0 ms (self)	0 %	
	🛚 🛃 Spring B	ean - paymentController:makePayment:83	16 ms (self)	0.1 %	
	🔻 🛃 Sprin	g Bean - org.springframework.transaction.interceptor.TransactionInterceptor#0:invoke:120	0 ms (self)	0 %	
	🔍 Sp	ring Bean - transactionManager:doCommit:657	28 ms (self)	0.1 %	1
	FB com.i	npetus.travelportal.util.BookCancelMail:bookingEmail:53	21091 ms (self)	99.8 %	11

Fig 16: Method call trace for a slow transaction