

StackStormによる イベントドリブンの 運用自動化

ブロードコムコミュニケーションズシステムズ株式会社

2016年11月30日



StackStormとは?

“If _____ Then _____”

StackStormとは?

ルール

“If _____ Then _____”

[トリガー]

disk.full

intf.down

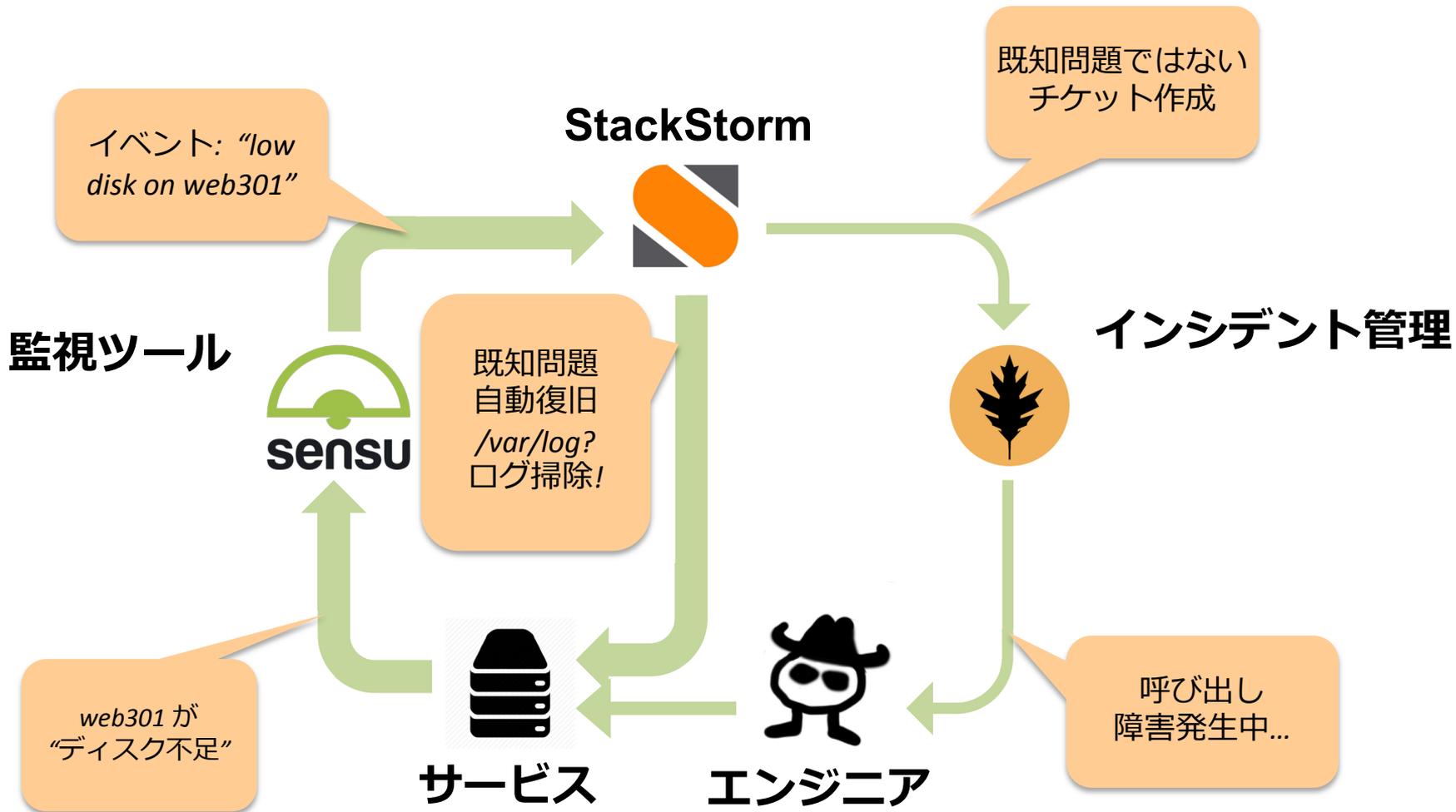
intf.still.dn

[アクション]

logrotate && rm -rf /tmp/*

ifconfig \$IFACE down && ...up

ssh \$switch; shut && no shut \$intf



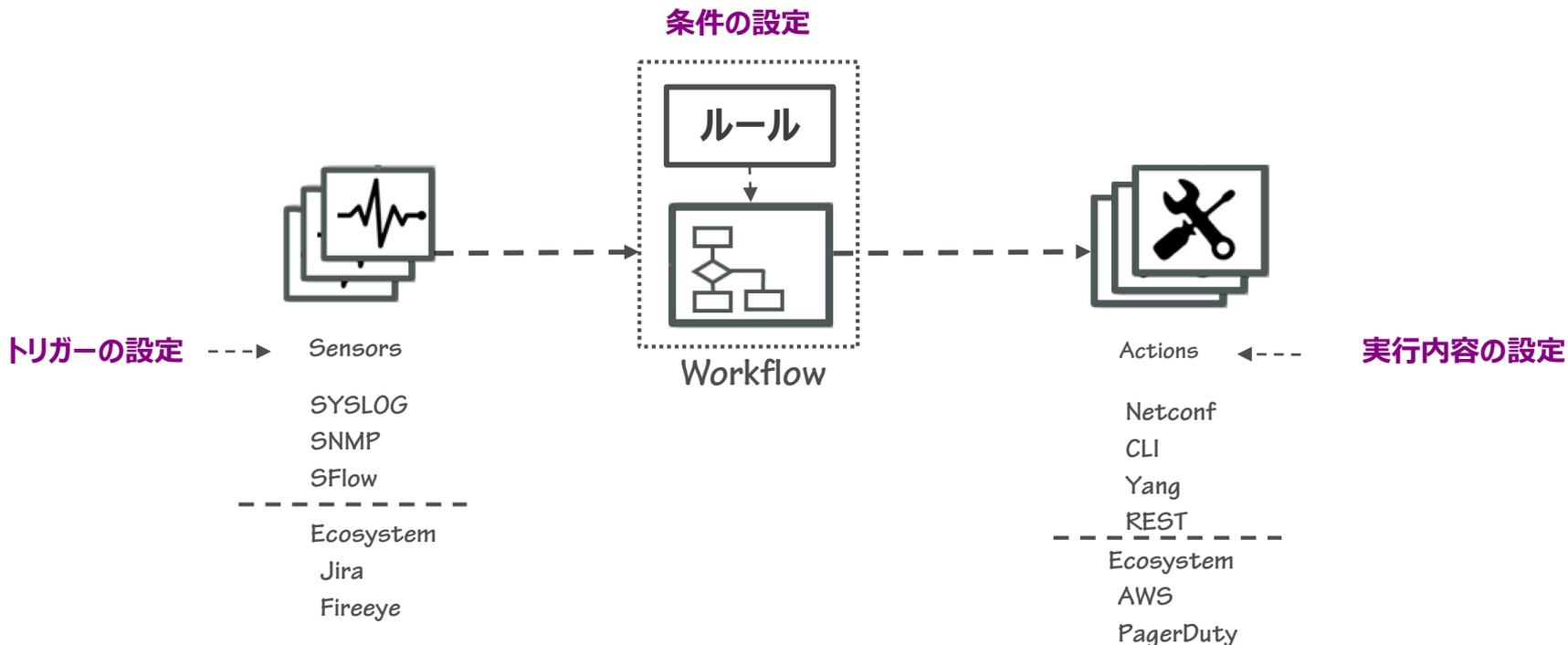
自動化対象の例

- セキュリティ対策
 - VMでマルウェア検出。VMをネットワークから隔離
- デプロイメント自動化“Blue-green app deployment”
 - Jenkinsテストをパス。新VMクラスタ作成、Appのインストールと設定、LBからトラフィックを流す、監視=>続行 or 切り戻し
- ネットワーク
 - BGPピアダウン。ログ収集、Slackへ通知&JIRAチケット作成
 - LAGメンバー障害。分散状況チェック、残りのLAGがあれば、障害リンクをDisable
- OpenStack orphan VM クリーンアップ
 - 非管理VMを検出。シャットダウン、オーナーへのメール通知、数日保管後、削除
- NFV: Nokia、AT&TではMistralとOpenStack
- OpenStack、ハードウェア障害時のVM退避
- Cassandra “node down” 自動復旧

StackStormのしくみ

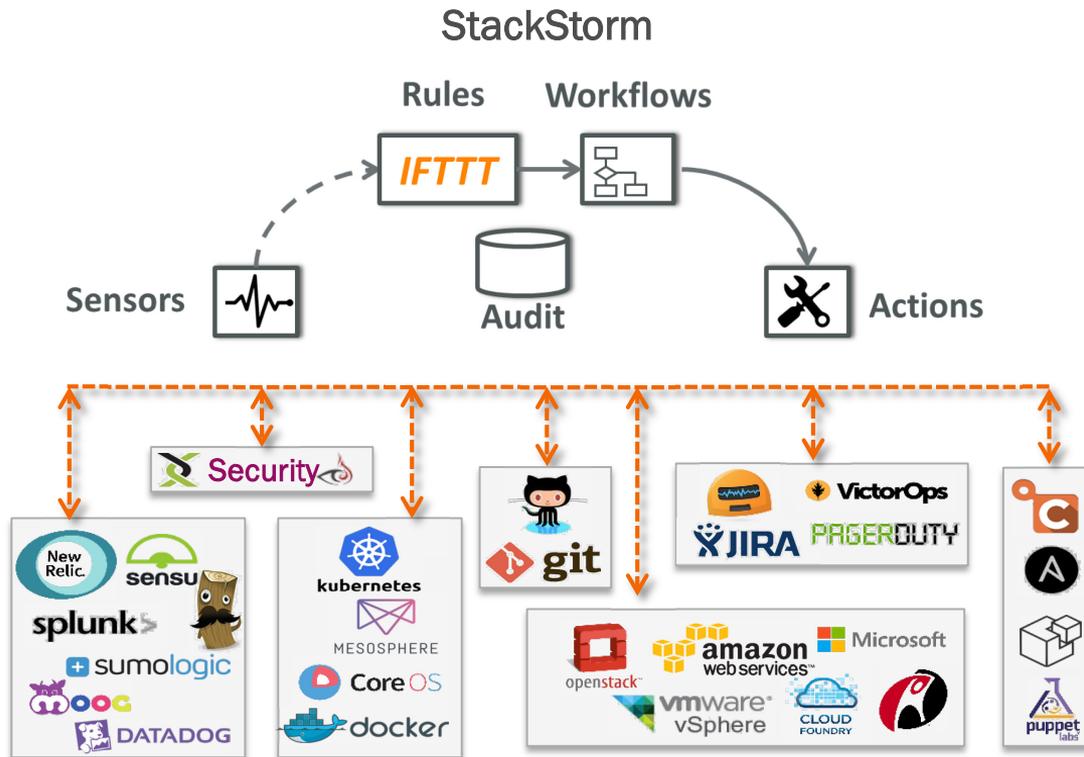
各コンポーネント

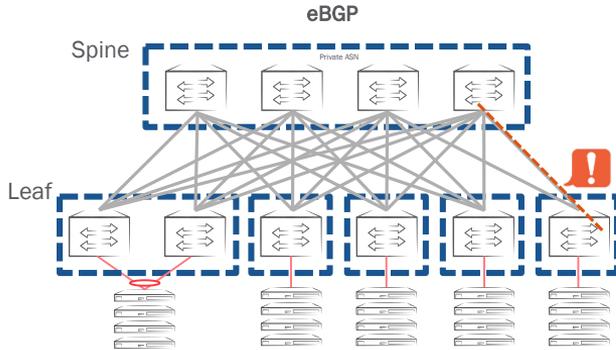
ルール : If_トリガー_ Then_ワークフロー_



クロスドメイン対応

ネットワークだけでない幅広い適応





StackStormアーキテクチャ

ワークフロー



ユーザ
インタフェース



構成可能な
サービス



Credential /
Authorization



Workflow
Engine



Device
Inventory



Data Store



Rules
Engine



REST API



Audit

Message Bus

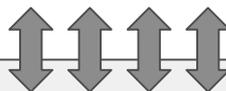
センサー
/アクション



Sensor



Sensor



Action



Action

Systems

Devices

Tools

BROCADE 

vmware
vSphere

openstack

docker

CLOUD
FOUNDRY

NUTANIX

amazon
webservices

Microsoft

JIRA

PAGERDUTY



StackStorm

オープンソース・プロジェクト

オープンソース・プロジェクト:

- BWC/ST2プラットフォーム
- ウェブUI
- セキュア・データ・ストア
- コミュニティ・パック

コミュニティ・サポート

Slackチャネルでサポート。コミュニティ版をGitHub経由で利用可能。コミュニティへのPull Requestを受け付け。

StackStorm



Brocade Workflow Composer

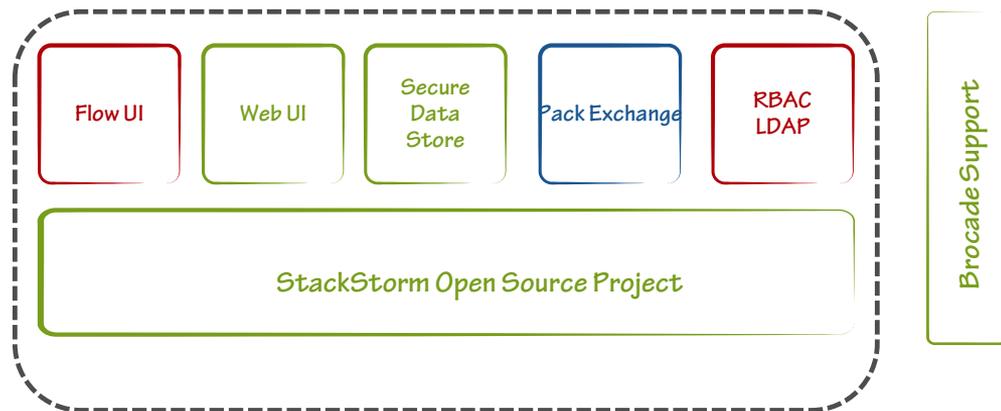
旧StackStorm Enterprise (商用版)

Workflow Composerのバンドル:

- StackStormプラットフォーム
- ウェブUI
- セキュア・データ・ストア
- **フローUI** - ワークフローをデザインするためのグラフィカル・インターフェース
- **ロール・ベース・アクセス・コントロール**と**LDAP**の統合
- **BWC Exchange** *コミュニティとサブスクリプションのパック管理

Brocade テクニカル・サポート

Brocade Workflow Composer



柔軟なWorkflowのカスタマイズ

直感的な操作感

```
1 version: '2.0'
2
3 cassandra.replace_host:
4   description: A basic workflow that replaces a dead cassandra node with a spare.
5   type: direct
6   tasks:
7     - dead_node
8     - replacement_node
9     - healthy_node
10
11 output:
12   just_output_the_whole_workflow_context: "<${ $%}"
13
14 tasks:
15   is_seed_node:
16     action: cassandra.is_seed_node
17     input:
18       nodes: "<${ $healthy_node %}"
19       node_id: "<${ $dead_node %}"
20     publish:
21       seed_node: "<${ $is_seed_node.get($healthy_node).stdout %}"
22     on-success:
23       - abort_replace: -<${ $seed_node = "True" %}"
24       - create_vm: -<${ $seed_node = "False" %}"
25     on-error:
26       - error_seed_node_determination: "<${ not $seed_node in list(false, True) %}"
27
28 abort_replace:
29   <[10]: 11
30   action: slack_post_message
31   input:
32     channel: "#demo"
33     message: "'[CASS-REPLACE-HOST] [<${ $dead_node %}] STATUS: FAILED REASON: %'"
34   on-complete:
35     - fail
36   error_seed_node_determination:
37     <[20]: 21
38     action: slack_post_message
39     input:
40       channel: "#demo"
41       message: "'[CASS-REPLACE-HOST] [<${ $dead_node %}] STATUS: FAILED REASON: %'"
42     on-complete:
43       - fail
44
45 create_vm:
46   <[10]: 11
47   action: core.local
48   input:
49     cmd: "echo Replacing <${ $dead_node%} with <${ $replacement_node %}"
50   on-success:
51     - stop_cassandra_service
52   on-error:
53     - notify_create_vm_failed
54
55 notify_create_vm_failed:
56   <[10]: 11
57   action: slack_post_message
58   input:
59     channel: "#demo"
60     message: "'[CASS-REPLACE-HOST] STATUS: FAILED REASON: create_vm_with_role'"
61   on-complete:
62     - fail
63
64 stop_cassandra_service:
```

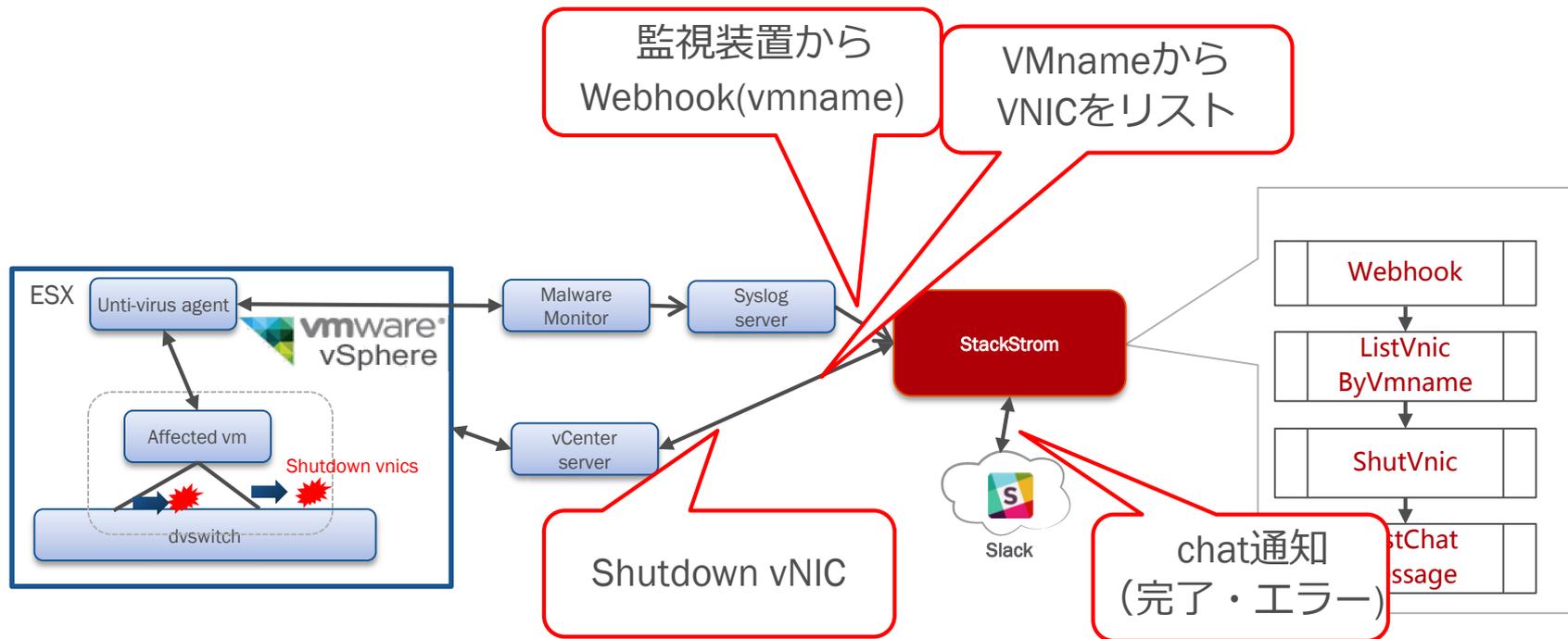
・ワークフローをGUI上で
作成、編集



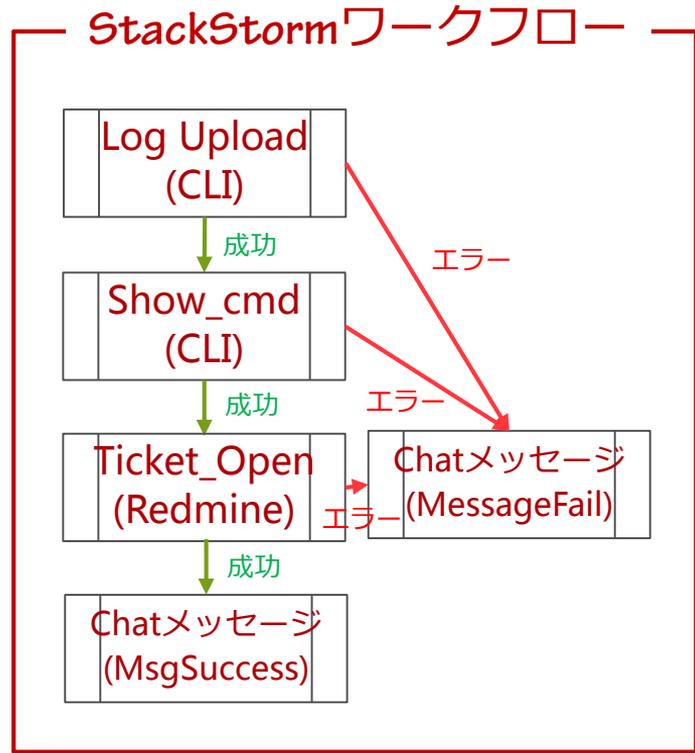
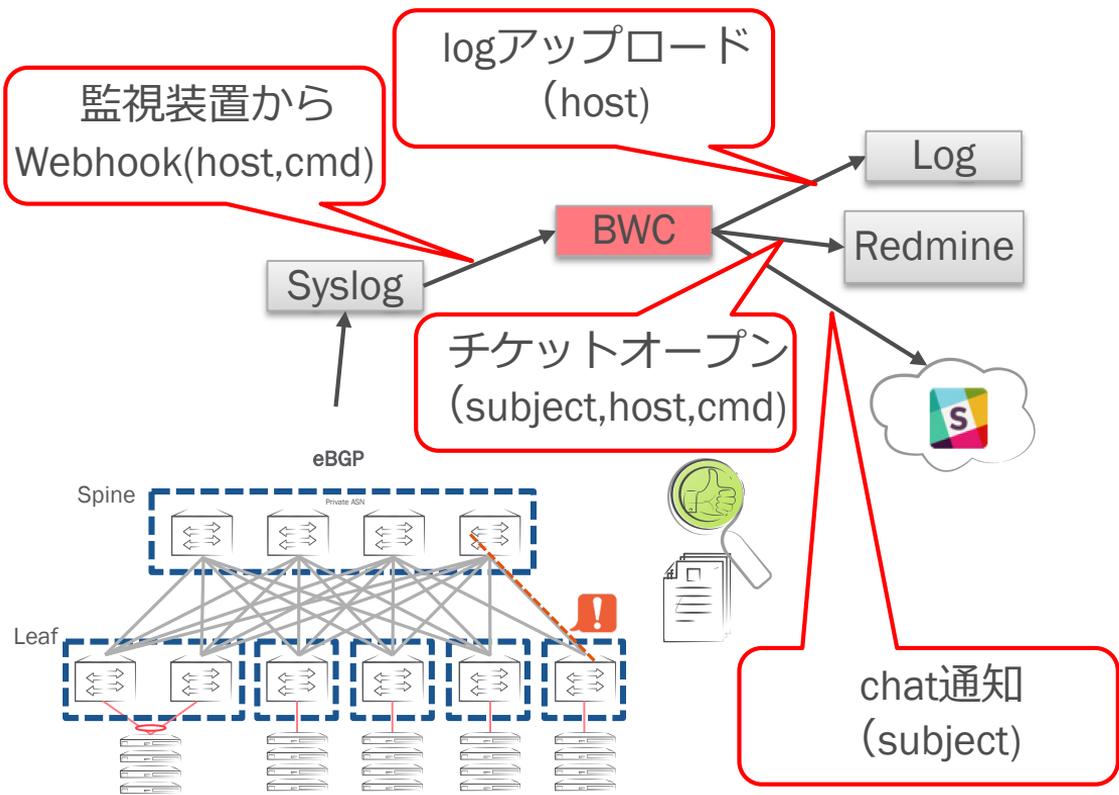
ユースケース

セキュリティ対策

ウィルス拡散防止



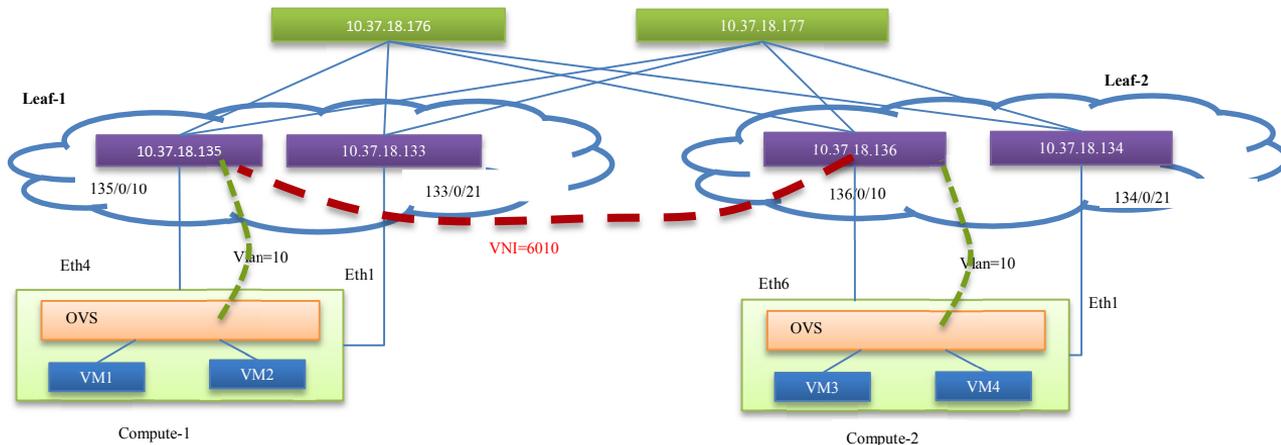
ログ収集 & チケットオープン自動化



Brocade OpenStack Neutron ML2 Automation Pack (デモ版)

<https://github.com/StackStorm/demo-neutron-bwc-openstack2016>

- オーバーレイネットワークの管理
- VDXのHW VTEPによるハイパフォーマンスVXLAN
- VMライフサイクルに連動したVXLANの自動作成・削除



BWC/StackStormシステム要件

BWC/StackStorm v2.0.1

Linux (64 bit)	Vagrant Box	Amazon AWS AMI
Ubuntu 14.04	bento/ubuntu-14.04	Ubuntu Server 14.04 LTS (HVM)
RHEL 7 / CentOS 7	bento/centos-7.2	Red Hat Enterprise Linux (RHEL) 7.2 (HVM)
RHEL 6 / CentOS 6	bento/centos-6.7	Red Hat Enterprise Linux (RHEL) 6 (HVM)

Testing	Production
<ul style="list-style-type: none">• Dual CPU system• 1GB RAM• 10GB storage• Recommended EC2: t2.medium	<ul style="list-style-type: none">• Quad core CPU system• >16GB RAM• 40GB storage• Recommended EC2: m4.xlarge

パッケージインストーラ (例: CLICRUD)

The screenshot displays the Brocade Workflow Composer interface. The main navigation bar includes 'BROCADE Workflow composer', 'HISTORY', 'ACTIONS', 'RULES', and 'DOCS'. The 'ACTIONS' menu is expanded, showing a list of actions: ANSIBLE, BWC-IPFABRIC, BWC-TOPOLOGY, CHATOPS, and CLICRUD. The 'CLICRUD' action is selected and expanded, showing its configuration options: clicrud.chain_remediate, clicrud.config_command, clicrud.ops_command, and register. The 'clicrud.ops_command' option is highlighted in orange. On the right side, the configuration details for 'clicrud.ops_command' are shown, including a 'GENERAL' tab and a 'PARAMETERS' section with fields for 'host *', 'command *', and 'env'. The 'packs.install' action configuration is also visible, showing a 'GENERAL' tab and a 'PARAMETERS' section with a 'packs *' field containing 'clicrud'.

URL: <https://172.24.199.58/#/actions/packs.install/general>

BROCADE Workflow composer

HISTORY ACTIONS RULES DOCS

Actions Filter

PACKS

packs.check_auto_deploy_repo
Check if a given branch in a ST2 Pack's Git repository should b... python-script

packs.delete
Deletes the pack from local content repository. python-script

packs.deploy
Deploy StackStorm Pack(s) mistral-v2

packs.download
Downloads packs and places it in the local content repository. python-script

packs.expand_repo_name
For supplied ST2 pack Repo return the Git URL and if it has a s... python-script

packs.info
Get currently deployed pack information python-script

packs.install
Installs packs from st2-contrib into local content repository. Wi... action-chain

packs.install
Installs packs from st2-contrib into local content repository. Wi... GENERAL

PARAMETERS

skip_notify

List of tasks to skip notific

env

Optional environment vari

packs *
clicrud

register

BROCADE Workflow composer

HISTORY ACTIONS RULES DOCS

Actions Filter

ANSIBLE

BWC-IPFABRIC

BWC-TOPOLOGY

CHATOPS

CLICRUD

clicrud.chain_remediate
Simple chain that when triggered by a webhook, will send info t... action-chain

clicrud.config_command
Login in to a Brocade ICX/MLX and run a configuration comma... python-script

clicrud.ops_command
Login in to a Brocade ICX/MLX and run an operational comma... python-script

GENERAL

PARAMETERS

host *
Valid IPv4 address or hostname that res

command *
Command for the operation 'show'

env
Environment variables which will be avai
key1=val1,key2=val2)

timeout

ワークフローサンプル

BROCADE[®] Workflow Designer

Current: default.iw2016 ⚙

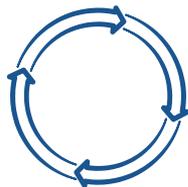
The image displays the BROCADE Workflow Designer interface. On the left, a flowchart shows a sequence of tasks: 'log_upload' (clicrud.ops_com...) connects to 'sh_cmd' (clicrud.ops_com...), which connects to 'ticket_open' (default.NewTicket). From 'ticket_open', the flow branches into two paths: one leading to 'ok_message' (slack.chat.postM...) and another leading to 'error_message' (slack.chat.postM...). Green arrows indicate successful execution paths, while red arrows with 'X' marks indicate error paths. On the right, a code editor shows the corresponding workflow definition in a structured text format.

```
1 ---  
2 version: '2.0'  
3  
4 default.iw2016:  
5 input:  
6 - host  
7 - command  
8 - subject  
9 tasks:  
10 log_upload:  
11 # [255, 26]  
12 action: clicrud.ops_command  
13 input:  
14 host: <% $.host %>  
15 command: "copy running-config scp://root:password  
16 on-success:  
17 - sh_cmd  
18 on-error:  
19 - error_message  
20 sh_cmd:  
21 # [185, 128]  
22 action: clicrud.ops_command  
23 input:  
24 host: <% $.host %>  
25 command: <% $.command %>  
26 publish:  
27 description: '<% task(sh_cmd).result.result %>  
28 on-success:  
29 - ticket_open  
30 on-error:  
31 - error_message  
32 ticket_open:  
33 # [115, 230]  
34 action: default.NewTicket  
35 input:  
36 subject: <% $.subject %>  
37 description: <% $.description %>  
38 on-success:  
39 - ok_message  
40 on-error:  
41 - error_message
```

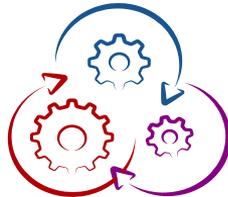
まとめ

StackStorm/BWCを是非一度お試しく下さい

イベント・ドリブンの
ライフサイクル



クロス・ドメイン
インテグレーション



柔軟なワークフロー
オープン



DevOps志向



Thank you