

IT Automation 収集機能・比較機能 【実習編】

※本書では「Exastro IT Automation」を「ITA」として記載します。

Exastro IT Automation Version 1.10 Exastro developer

目次

はじめに

(1) 本書について
 (2) 作業環境
 (3) シナリオ

1. 実習1【収集機能】ターゲットホストのOS情報を収集する

 実習1全体図

 1.1 ターゲットホストの登録

 1.2 オペレーションの登録

 1.3 Movementの登録

 1.3.1 ヘッダーセクションとgather facts

 1.4 Playbookの登録

 1.4.1 YAMLファイルと収集用ディレクトリ

 1.5 Movement-Playbook紐付

 1.6 作業対象ホストの登録

 1.7 収集値を登録するパラメータシートの作成

 1.8 収集項目値管理の登録

 1.9 収集インターフェース情報の登録

 1.10 作業実行

 1.11 収集結果の確認

2. 実習2 【比較機能】実習1で収集した値と期待値を比較する

<u>実習2全体図</u> <u>2.1 オペレーションの登録</u> <u>2.2 期待値用パラメータシートの作成</u> <u>2.3 期待値の登録</u> <u>2.4 比較定義の登録</u> <u>2.5 比較実行</u> <u>【参考】比較定義詳細</u> 【参考】(1)比較定義の登録 目次

【参考】 (2) 比較定義詳細の登録 【参考】 (3) 比較実行

3. 実習3【収集機能】ターゲットホストのSSL証明書ファイルを収集する

 実習3全体図

 3.1 ターゲットホストの登録

 3.2 オペレーションの登録

 3.3 Movementの登録

 3.4 Playbookの登録

 3.4 Playbookの登録

 3.5 Movement-Playbook紐付

 3.6 ファイル名の登録

 3.7 代入値自動登録設定の登録

 3.8 収集値を登録するパラメータシートの作成

 3.9 収集項目値管理の登録

 3.10 収集インターフェース情報の登録

 3.11 作業実行

 3.12 収集結果の確認

4. 実習4 【比較機能】実習3で収集したSSL証明書ファイルを、異なる日時 に収集したファイルと比較する

<u>実習4全体図</u> <u>4.1 オペレーションの登録</u> <u>4.2 差分有りSSL証明書の用意</u> <u>4.3 ファイル名の登録</u> <u>4.4 作業実行</u> <u>4.5 収集結果の確認</u> <u>4.6 比較定義の登録</u> <u>4.7 比較実行</u>

はじめに



(1) 本書について



本書では「収集機能」と「比較機能」について、実習形式で作業を進め、理解を深めていただけます。



(2) 作業環境



本書で使用する作業環境は以下の通りです。



※1 ITAはRHEL7系およびRHEL8系のOSで導入いただけます。 ※2 Ansibleの動作対象となれるOSであれば、問題なくご利用いただけます。

パラメータの収集・比較とファイルの収集・比較

- ●実習1と2ではパラメータの収集・比較、実習3と4ではファイルの収集・比較を解説 しています。
- ●具体的な収集対象は「パラメータ:OS情報」「ファイル:SSL証明書ファイル」となり ます。

	収集機能	比較機能
パラメータの 収集・比較	実習1 ターゲットホストのOS情報 を収集する	実習2 実習1で収集した値と期待値 を比較する
ファイルの 収集・比較	実習3 ターゲットホストのSSL証明 書ファイルを収集する	実習4 実習3で収集したSSL証明書 ファイルを、異なる日時に収 集したファイルと比較する

1. 実習1【収集機能】ターゲットホストのOS情報 を収集する



実習1全体図

実習1の作業の流れ

- ●数字は本書の章番号です。
- ●各種設定を行ってから作業実行し、インベントリ(OS情報)を収集してパラメータシー トへ自動登録します。



1.1 ターゲットホストの登録

ターゲットホストの接続情報を登録する

「機器一覧」から登録します。

メニュー: 基本コンソール > 機器一覧

① 登録 > 登録開始を押下する。

② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。



HW ホス 機器種別 (任意)		IPアドレス (ご使用の環境 に応じて)	ログイン ユーザID (ご使用 の環境に 応じて)	ログィ	(ンパスワード	Ansible利用情報
	ホスト名 (任意の名称)			管理	ログイン パスワード (ご使用の環境	Legacy/Role利用 情報
					に応じて)	認証方式
SV	targethost	192.0.2.1	root	•	******	パスワード認証

1.2 オペレーションの登録

今回のオペレーションを登録する

オペレーションはITAの自動作業一式を指します。 この後、関連するすべてのデータをオペレーションに紐付けていきます。

メニュー:基本コンソール > オペレーション一覧

① 登録 > 登録開始を押下する。

② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。



Ansible-LegacyのMovementを登録する

Movementとは、ITAの自動作業の最小単位で、ジョブを表します。 この後Playbookに紐付けて、OS情報を収集するMovementとなります。

- メニュー: Ansible-Legacy > Movement一覧
- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

MovementID Movement名		遅延タイマー	Ansible ^我 用情報			
			ホスト指定形式*	WinRM接続	ヘッダーセクション	
自動入力	GatherFacts		IP v	•	- hosts: all remote_user: "{{loginuser }}" gather_facts: yes become: yes	

Movement名		Ansible利用情報	
(任意の名称)	ホスト指定形式	ヘッダーセクション	(gather_factsを有効にし
GatherFacts	IP	<pre>- hosts: all remote_user: "{{loginuser }}" gather_facts: yes become: yes</pre>	ます。 ※ 詳細は <u>『1.3.1 ヘッダーセク</u> ションとgather_facts』参照)

1.3.1 ヘッダーセクションとgather_facts

gather_factsを有効にする

- ITAのデフォルトの設定では、AnsibleのPlaybookのヘッダーセクションのgather_facts が無効になっています。
- ●この実習ではgather_factsを使ってOS情報を取得するため、ヘッダーセクションに下記のように入力して、有効にしておきます。
- ●デフォルト値から変更する必要がない場合、ヘッダーセクションは空欄でOKです。



作業実行用Playbookの作成(1/2)

- ●「ターゲットホストの/tmp/直下に、取得したOS情報を記載したYAMLファイルを生成 → 生成したYAMLファイルをITAホストサーバの収集ディレクトリにコピーする」という内容です。
- YAMLファイルと収集用ディレクトリについては、<u>『1.4.1 YAMLファイルと収集用ディレクトリ』</u>を参照してください。



1.4 Playbookの登録(2/3)

作業実行用Playbookの作成(2/2)

- name: make yaml file blockinfile: create: ves mode: 0644 insertbefore: EOF marker: "" dest: "/tmp/gatherfacts.yml" content: ansible architecture : {{ ansible_architecture }} : {{ ansible bios version }} ansible_bios_version ansible default ipv4 address : {{ ansible default ipv4.address }} ansible default ipv4 interface : {{ ansible default ipv4.interface }} ansible default ipv4 network : {{ ansible default ipv4.network }} ansible distribution : {{ ansible_distribution }} ansible_distribution_file_path : {{ ansible_distribution_file_path }} ansible distribution file variety : {{ ansible distribution file variety }} ansible distribution major version: {{ ansible distribution major version }} ansible_distribution_release : {{ ansible_distribution_release }} ansible distribution_version : {{ ansible_distribution_version }} : {{ ansible_machine }} ansible machine : {{ ansible_memtotal mb }} ansible memtotal mb ansible nodename : {{ ansible nodename }} ansible_os_family : {{ ansible_os_family }} ansible_pkg_mgr : {{ ansible_pkg_mgr }} : {{ ansible_processor_cores }} ansible_processor_cores - name: copy the make yaml file to local fetch: src: "/tmp/gatherfacts.yml" dest: "{{ ___parameter_dir___}}/{{ inventory_hostname }}/"

ファイル名:GatherFacts.yml

flat: yes

Ansible-LegacyにPlaybookを登録する

作成した作業実行用Playbookを登録します。

メニュー: Ansible-Legacy > Playbook素材集

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

1.4.1 YAMLファイルと収集用ディレクトリ(1/2)

YAMLファイルを収集用のディレクトリに生成する

- ●ITAでは、収集結果ファイルはYAML形式に指定されているため、YAMLファイルを生成 する必要があります。
- ●生成されたYAMLファイルは、ITAの予約変数で指定された収集用ディレクトリに格納します。

dest: "{{par	rameter_dir }}/{{ inver	GatherFacts.yml 下から2行目 htory_hostname }}/"
パスの変数	このデー	ィレクトリが指定されている
	ITA予約変数	変数指定内容
ソースファイルの 格納先	parameter_dir	作業結果ディレクトリ配下の「_parameters」のパス
収集したファイルの 格納先	parameters_file_dir	作業結果ディレクトリ配下の「_parameters_file」のパス

1.4.2 YAMLファイルと収集用ディレクトリ(2/2)

収集用のディレクトリは、下記のようなファイルの階層構造になっています。

ファイルの階層構造



MovementとPlaybookを紐付ける

登録したMovementとPlaybookを紐付けます。

メニュー: Ansible-Legacy > Movement-Playbook紐付

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。



1.6 作業対象ホストの登録

オペレーション・Movement・作業対象ホストを紐付ける

登録したオペレーション・Movement・作業対象ホストを紐付けます。

メニュー: Ansible-Legacy > 作業対象ホスト

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

項番	オペレーション*			Movement*	ホスト*
7	7 6:GatherFacts1		[4:GatherFacts	2:targethost 🔹
オペ	レーション	Movement	ホスト		
Gat	herFacts1	GatherFacts	targethos	t	

1.7 収集値を登録するパラメータシートの作成(1/4)

収集した値を登録するパラメータシートを作成する

[Gathered Facts] メニューを作成します。これがパラメータシートとなり、収集した値 が自動登録されます。

メニュー:メニュー作成>メニュー定義・作成

① 各項目で下表のように選択または入力する。

② [作成] ボタンを押下する。



1.7 収集値を登録するパラメータシートの作成(2/4)

1.基本情報

メニュー名(任意の名称)	作成対象	表示順序
Gathered Facts	パラメータシート(ホスト/オペレーションあり)	1

2.対象メニューグループ

入力用	代入値自動登録用	参照用
入力用(デフォルト)	代入値自動登録用(デフォルト)	参照用(デフォルト)

3.項目

項目名(任意の名称)	入力方式	最大バイト数(任意の値)
ansible_architecture	文字列(単一行)	128
ansible_bios_version	文字列(単一行)	128
ansible_default_ipv4 > address (<u>*</u>)	文字列(単一行)	128
ansible_default_ipv4 $>$ interface (\times)	文字列(単一行)	128
ansible_default_ipv4 > network (<u>*</u>)	文字列(単一行)	128
ansible_distribution	文字列(単一行)	128
ansible_distribution_file_path	文字列(単一行)	128
ansible_distribution_file_variety	文字列(単一行)	128
ansible_distribution_major_version	文字列(単一行)	128
ansible_distribution_release	文字列(単一行)	128

1.7 収集値を登録するパラメータシートの作成(3/4)

項目名(任意の名称)	入力方式	最大バイト数(任意の値)
ansible_distribution_version	文字列(単一行)	128
ansible_machine	文字列(単一行)	128
ansible_memtotal_mb	文字列(単一行)	128
ansible_nodename	文字列(単一行)	128
ansible_os_family	文字列(単一行)	128
ansible_pkg_mgr	文字列(単一行)	128
ansible_processor_cores	文字列(単一行)	128

※ カラムグループ作成

ansible_default_ipv4 > address
ansible_default_ipv4 > interface
ansible_default_ipv4 > network

[ansible_default_ipv4] カラムグルー プを作成し、その中に [address] [interface] [network] の3カラムを 入れ込みます。

	L	ansible_d	efault_ipv4			
address		interface			network	
文字列(単一行)	~	文字列(単一行)) 🗸	<u>¢</u>	文字列(単一行) 🗸 🗸	
最大バイト数* 128		最大バイト数*	128	最大	いイト数*	128
正規表現		正規表現		ī	E規表現	
初期値		初期値			初期値	
□ 必須 □ 一意制約		□ 必須 □ 一意制約			□必須 □ 一意制約	
影光印列		影之同日		202.09 202.09		比미
借考	信守 信守		17			

1.7 収集値を登録するパラメータシートの作成(4/4)



収集項目値管理を登録する

- 収集項目(FROM)のYAMLファイル名・変数名と、パラメータシート(TO)のメ ニュー名・項目名を紐付けます。
- ●下表の通り、各変数と項目を1セットとして登録します。

メニュー: Ansible共通 > 収集項目値管理

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

収集項目(FROM)

パラメータシート(TO)

	(\mathcal{C}	
TD		収集項目(F	ROM)		パラメータシート(TO)	
10	パース形式*	PREFIX(ファイル名)*	変数名*	メンパ変数	メニューグループ:メニュー*	項目*
自動入力	YAML 🔻	gatherfacts	ansible_architectu		2100011611:代入値自動登録用:2:Gathered Facts ▼	パラメータ/ansible_architecture 🔹

1.8 収集項目値管理の登録(2/3)

	収集項目(FROM)		パラ	メータシート(TO)
パース 形式	PREFIX (ファイル名)	変数名	メニューグループ :メニュー	項目
YAML	gatherfacts	ansible_architecture	代入値自動登録用: Gathered Facts	パラメータ/ ansible_architecture
YAML	gatherfacts	ansible_bios_version	代入値自動登録用: Gathered Facts	パラメータ/ ansible_bios_version
YAML	gatherfacts	ansible_default_ipv4address	代入値自動登録用: Gathered Facts	パラメータ/ ansible_default_ipv4/address
YAML	gatherfacts	ansible_default_ipv4interface	代入値自動登録用: Gathered Facts	パラメータ/ ansible_default_ipv4/interface
YAML	gatherfacts	ansible_default_ipv4network	代入値自動登録用: Gathered Facts	パラメータ/ ansible_default_ipv4/network
YAML	gatherfacts	ansible_distribution	代入値自動登録用: Gathered Facts	パラメータ/ ansible_distribution
YAML	gatherfacts	ansible_distribution_file_path	代入値自動登録用: Gathered Facts	パラメータ/ ansible_distribution_file_path
YAML	gatherfacts	ansible_distribution_file_variety	代入値自動登録用: Gathered Facts	パラメータ/ ansible_distribution_file_variety
YAML	gatherfacts	ansible_distribution_major_version	代入値自動登録用: Gathered Facts	パラメータ/ ansible_distribution_major_ver sion
YAML	gatherfacts	ansible_distribution_release	代入値自動登録用: Gathered Facts	パラメータ/ ansible_distribution_release

1.8 収集項目値管理の登録(3/3)

	収	集項目(FROM)	パラン	メータシート(TO)
パース 形式	PREFIX (ファイル名)	変数名	メニューグループ :メニュー	項目
YAML	gatherfacts	ansible_distribution_version	代入値自動登録用: Gathered Facts	パラメータ/ ansible_distribution_version
YAML	gatherfacts	ansible_machine	代入値自動登録用: Gathered Facts	パラメータ/ ansible_machine
YAML	gatherfacts	ansible_memtotal_mb	代入値自動登録用: Gathered Facts	パラメータ/ ansible_memtotal_mb
YAML	gatherfacts	ansible_nodename	代入値自動登録用: Gathered Facts	パラメータ/ ansible_nodename
YAML	gatherfacts	ansible_os_family	代入値自動登録用: Gathered Facts	パラメータ/ ansible_os_family
YAML	gatherfacts	ansible_pkg_mgr	代入値自動登録用: Gathered Facts	パラメータ/ ansible_pkg_mgr
YAML	gatherfacts	ansible_processor_cores	代入値自動登録用: Gathered Facts	パラメータ/ ansible_processor_cores

1.9 収集インターフェース情報の登録

収集インターフェース情報を登録する

収集した値をITAのパラメータシートに登録する際のREST APIアクセスで必要になるため、RESTユーザー/パスワードを実行権限のあるユーザーで登録します。

メニュー: Ansible共通 > 収集インターフェース情報

- ① [フィルタ]を押下する。
- 「一覧」に1行だけ表示されるので、 [更新] ボタンを押して下表のように入力し、
 [登録] ボタンを押下する。

履歴	更新	ID⇔	ホスト名合	IP≙	RESTユーザー合	RESTパスワード	REST方式合	プロトコル会	ポート会	アクセス権
	~~~~									アクセス許可ロール⇔
履歴	更新	1	localhost	127.0.0.1	administrator	*****	IP	http	80	
	•									
ID	ホスト名	'*	IP*	RESTユーサ	ー RESTパスワード	REST方式*	プロトコ	コル* ポート		
1 100	alhost		127.0.0.1	administrato	· [		http	80		
	RES	тд-	-ザー	R	ESTパスワート	\$				
実行	権限	のある	3ユーザー	そのユ	ーザーのパスワ	フード				

# 1.10 作業実行(1/2)



Movementとオペレーションを選択し、作業実行します。

#### メニュー: Ansible-Legacy > 作業実行

- ① Movement[一覧] から登録したMovementを選択する。
- ② オペレーション[一覧]から登録したオペレーションを選択する。
- ③ [実行] ボタンを押下する。

Movement[一覧]		
Revenent10e Rovenent2e オークストレータe 遅延タイマー会 水スト版定形式 winan45kg ペロダーセクションタ オブションバラメータe virtualeve ansi     an	ble.cfg アクセス株 アクセス株可ロール。 第9 2022/05/10 10:05:0 データボー	R林交新者会 タビリティブロシーショ
gather_facts: yes hermer vas フィルク結果作数: 2	2022/05/23 10:44:20 データボー	タビリティブロシージャ
オペレーション(フィルタ)		
オペレーション(一覧) 選択 No.0 オペレーション10 き オペレーション名き 実施予定日時 - 最終実行日時 - アクセス度 - 最考会 最終更新日時 - 最終更新日時 - 最終更新日時 - 最終更新日時		
・         5         5         6 datherFacts1         2021/04/22         17:09         2022/05/23         13:40         2022/05/23         13:40:02         LegacyPT業実行プロシージャ           0         6         6 datherFacts2         2021/04/23         17:10         2022/05/23         13:13:14         7 - ッホーッとUJT+J-UJンーシャ           7         7         7 get5511         2021/04/23         17:10         2022/05/23         11:16:00         1egacyPT業実行プロシーシャ           8         8         get5512         2021/04/28         12:19         2022/05/23         11:43:00         2022/05/23         1egacyPT業実行プロシーシャ		
フィル94編作数: 4 MovementD 6	Movement[一覧]	オペレーション[一覧]
Movement & GatherFacts	GatherFacts	GatherFacts1

# 1.10 作業実行(2/2)

### 作業のステータスを確認する

[作業状態確認] メニューで、ステータスが完了になれば、作業完了です。

メニュー: Ansible-Legacy > 作業状態確認

· Menu 対象	作業			
ovement-5		項目		值
作業N	ю.			22
pook素材集	重別			通常
ステー	ータス			完了
ement-Playbook紐付 実行コ	エンジン			Ansible Core
空—四 空—四	Symphony			
行 <u>见</u> 呼出元	Conductor			
值自動登錄設定 実行二	ユーザ			システム管理者
		ID		6
対象ホスト		名称		GatherFacts
值管理		遅延タイマ(分)		
E B-1		Ancible利田信報	ホスト指定形式	IP
実行 Moven	ment	AU2TOTCJ.JUBHK	WinRM接続	
		Ansible Core利用情報	virtualenv	
状態確認		Ansible Tower利用情報	virtualenv	
管理		Ansible Automation Controller利用情報	実行環境	
		ansible.cfg		
		No.		5
オペレ	ノーション	名称		GatherFacts1

### 収集状況を確認する

収集の成功/失敗を確認します。

- メニュー: Ansible-Legacy > 作業管理
- ① [フィルタ] ボタンを押下する。
- ② 一覧 > 収集状況 > ステータス に、以下のように表示される。
  - 収集済み : 収集成功
  - 収集済み(通知あり) : 登録/更新中に不備があった場合
  - 対象外

: 収集失敗

・ 収集エラー

: 登録したオペレーションかターゲットホストの情報に不備が あった場合

							収集状況
履歴	作業No.⊜	作業状態確認	実行種別♦	ステータス令	実行エンジン♦	ステータス令	収集ログ
厢歴	22	作業状態確認	通常	完了	Ansible Core	収集済み	<u>CollectData_000000002.log</u>

# 1.11 収集結果の確認(2/2)

### パラメータシートを確認する

値がパラメータシートに登録されているか確認します。

#### メニュー: 入力用(もしくは参照用) > Gathered Facts

#### ① [フィルタ] ボタンを押下する。

② 一覧を表示し、作成した項目に値が入っているか確認する。

								オペレーショ	ョン			パラメ・	
履歴	複製	更新	廃止	No⇔	ホスト名⇔	ID⇔	オペレーション名⇔	基準日時令	実施予定日時令	最終実行日時令	ansible_architecture	ansible_bios_version🕀	
履歴	複製	更新	廃止	1	targethost	6	GatherFacts1	2021/07/28 16:47	2021/06/10 09:56	2021/07/28 16:47	x86_64	1.11.0-2.el7	
•••	-タ					ans	ible_default_ipv4				ancible dictributio	A ancible distribution file	nath≙
	ansib] 192.0.2	le_def 2.1	ault_	ipv4	> address	⇔ ans eth	sible_default_ipv4 0	> interface⇔	ansible_default_ 192.0.2.0	_ipv4 > networ	CentOS	/etc/redhat-release	
***												•••	
	ansib	)le_di	strib	ution	_file_varie	ty⊜	ansible_distribut	ion_major_versio	n⇔ansible_distr	<pre>`ibution_release</pre>	.⇔ ansible_distributio	n_version	
*	RedHat	t					7		Core		7.8	*	
•••													
	ansil	ble_ma	achine	e⇔ a	nsible_memt	otal_	mb⇔ ansible_	nodename⇔	ansible_os_famil	.y⇔ ansible_pkg	g_mgr⇔ ansible_process	or_cores⇔	
*•	x86_6	4		17	771		demo.localdom	ain	RedHat	yum	1		

# 2. 実習2 【比較機能】実習1で収集した値と期待値 を比較する



### 実習2の作業の流れ

● ITAに期待値を登録し、実習1で収集した値と比較します。



### 2.1 オペレーションの登録

### オペレーションを登録する

比較用のオペレーションを登録します。

#### メニュー:基本コンソール > オペレーション一覧

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

No	オペレーションの	オペレーショング*	宝炼子宁口哇*		アクセス権
NO.			天心了た口呀	設定	アクセス許可ロール
10	10	GatherFacts2	2021/10/01 09:25	設定	

オペレーション名	実施予定時間
(任意の名称)	(任意の時間)
GatherFacts2	2021/10/01 09:25

# 2.2 期待値用パラメータシートの作成(1/3)

### 期待値登録用のパラメータシートを作成する

実習1で作成した [Gathered Facts] メニューを複製し、「メニュー名」「表示順序」だけ変更します。

#### メニュー:メニュー作成>メニュー定義一覧

- 「フィルタ」を押下して「一覧/更新」に [Gathered Facts] メニューを表示したら、 「メニュー定義・作成」ボタンを押下する。
- ② メニュー定義の閲覧画面が表示されたら、 [流用新規] ボタンを押下する。
- ③ 「メニュー名」「表示順序」だけ空の状態で複製されるので、下表のように入力する。
- ④ [作成] ボタンを押下する。

ansible_machine						- Reddin : Reddin :
字列(単一行) 👻	文字列(単一行) ¥	文字列(単一行) イ	文字列(第一行) イ	文字列(單一行)	* 文字列(単一行) *	xiiii 12 xiiii = 8.1 Cathered Farts
リ(イト歌* 128	最大バイト数* 128	最大バイト歌* 128	最大バイト数* 128	最大バイト数* 128	最大バイト数* 128	Autoreu raus
E成表現	正規表現	正成表現	正映表現	正規表現	正規表現	>=>=>)
初期時間	814835	EDIRES	初期相	a second	204425	R04091 1
必須 〇一憲制約	○ 必須 ○ 一意制約	○心須 ○一意制約	○心須 ○一意制約	○必須 ○一意利約	○ 必須 ○ 一意制約	ホストグループ利用:
100						経メニュー利用:
						BallBeal(EM)   2022-01-11 11:47:54
						単純実形者:システム管理者
						2 - 208 47 3 - 40 - 1
						入力用: 入力用
						代入前自動空間用:代入後自動管理用
						PR01 : 240.0
				_		
ta- 07						#89/(#898)
一覧(プレビュー)						<i>1</i> (9->:
	<i>"</i>					
# 2.2 期待値用パラメータシートの作成(2/3)

項目 グループ	リピート 取り消し やり直し	J				
X sible_machine	× ansible_memtotal_mb	X	×	×	X ansible_processor_cores	メニュー作成情報
						項齋 自動入力
Ē-(T) <b>∨</b>	又子列(単一行) ▼	又子列(里一行) ♥	又子列(里一行) 🗸	又子列(単一行)    ▼		x⊐⊐−8*:
卜数* 128	最大バイト数* 128	最大バイト数* 128	最大バイト数* 128	最大バイト数* 128	最大バイト数* 128	作成対象: パラメータシート (ホスト/オペレーシ 🗸
現	正規表現	正規表現	正規表現	正規表現	正規表現	表示順序*:
<u>ā</u>	初期値	初期値	初期値	初期値	初期値	ホストグループ利用: ■ 利用する
□ 一意制約	□ 必須 □ 一意制約		□ 必須 □ 一意制約	□ 必須 □ 一意制約	□ 必須 □ 一意制約	利用する
彩之明日	151.00J	副注明	251.013	副注明月	101.019	最終更新日時: 自動入力
						最終更新者: 自動入力
偏考	信考	信号	信号	信号	借守	
						対象メニューグループ
						入力用*:入力用
						代入値自動登録用*: 代入値自動登録用
						参照用*: 参照用
4						対象メニューグループを選択
-						
						一意制約(複数項目)
一覧(プレビ・	л —)					パターン:
32(202-	- /					一意制約(複数項目)を選択
		オペレーション				- アクセス許可ロール
	<b>4</b> )				ansit	ロール:
	●ペレーション名令 基	準日時会 実施予定日会	最終実行日時会 ansible_a	rchitecture⊖ ansible_bio	os_version⇔ address⇔	アクセス許可ロールを選択
1 192.16	.0.1 オペレーション 2020/	01/01 00:00 2020/01/01 00:0	0 文字列(単一)	行) 文字列(単一行	) 文字列(単一行) :	
2 192.158	.0.1 オペレーション 2020/	01/01 00:00 2020/01/01 00:0	0 文字列(単一)	行) 文字列(単一行	) 文字列(単一行) :	- ICH)
∕E ΒΫ						
11900						
メニュ-	一名(仕恵の名	5杯)	表示順序		メニューの	表示順は
	OS情報		3		仕恵です。	

## 2.2 期待値用パラメータシートの作成(3/3)





### 期待値を登録する

作成した[OS情報]メニューに期待値を登録します。実習1で収集した値と差分を出しておきたいので、「ansible_default_ipv4__address」だけ異なる値を入力しておきます。

#### メニュー:入力用 > OS情報

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

ио <u></u>		^ب لا	ペレーション		パラメータ	
		<i>*</i> /	ペレーション*	insible_architecture ansible_b	ios_version ansible	_default_ipv4 > address
1 targethost v		2021/10/01 09:25_10:Gath	erFacts2 v	36_64 [1.11.0-2.ε	17 192.0.2.	2
				パラメータ/エ	頁目名	
ホスト名	77	レーション	ansible ipv4	_default_ _address	२०	の他の項目

### 比較対象となる2つのメニューを選択する

期待値として登録した値と、実習1で収集した値を比較します。

#### メニュー:比較 > 比較定義

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

No 比較定義名*	比較対象メニュー1	比較対象メニュー2	全件一致
2 <u>0S'情報</u> 2	2100011611:代入値自動登録用:100:OS情報 ▼	2100011611:代入値自動登録用:91:Gathered Facts 👻	
比較定義名 (任意の名称)	比較対象メニュー1	比較対象メニュー2	全件一致
OS情報	代入値自動登録用:OS情報	代入値自動登録用:Gathered Facts	•
	対 「 ※ 」	象メニューの項目すべてを比 ●」を選択する。	較するので <u> 定義詳細』</u> 参照

# 2.5 比較実行(1/2)

## 定義した比較を実行する

実習1で収集した値と期待値を比較します。

メニュー:比較 > 比較実行

① 比較実行 で、下表のように選択または入力し、 [比較] ボタンを押下する。
 ② 比較結果が表示される。

比較定義: 2:OS情報【100:OS情報 - 91:Gathered Fact… ▼ 基準日1: 基準日2: 対象ホスト: 選び 出力内容: ② 全件出力 ③ 差分のみ 比較	比較実行				
比較	比較定義: 2:05情報 【 100 出力内容:	):OS情報 - 91:Gathered Fact… 差分のみ	▼ 基準日1:	基準日2:	対象ホスト: 選択
	比較				



# 2.5 比較実行(2/2)



## パラメータシートの項目を限定して比較する

●項目を限定して比較する場合、「比較定義詳細」メニューを利用します。



### 比較対象となる2つのメニューを選択する

比較するメニューを選択します。項目を限定して比較するので「全件一致」はOFFにします。

#### メニュー:比較 > 比較定義

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

No	比較定義名	比較対象メニュー1	比較対象メニュー2	全件一致
自動入力	IPアドレス	2100011611:代入値自動登録用:211:OS情報 ▼	2100011611:代入値自動登録用:208:Gathered Facts 👻	•

※*は必須項目です。



## 比較対象となる項目を選択する

IPアドレス

選択したメニューの中から、比較する項目を選択します。

### メニュー:比較 > 比較定義詳細

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

OS情報:パラメータ/

ansible default ipv4/address

No	比較調	主義名*	表示項目名*		対象カラム1	•
	IPアドレス [ 100:OS信報-91:Ga	thered Facts ] 💌	IPアドレス	【IPアドレス2100011611:代入値自動登録用:100:OS情報:129:パラメータ/ansible_default_ipv4/ansibl		ible_default_ipv4… 💌
			2100011611:{{	入値自動登録用:91:Gathered F	対象カラム2 acts:109:パラメータ/ansible_default_ipv4/ansible_defaul	表示順
	比較定義名	表示項目名	対象力	ラム1	対象カラム2	表示順
IF	マアドレス		代入値自動登録用:		代入値自動登録用:	

Gathered Facts:パラメータ/

ansible default ipv4/address

[ Gathered Facts-

OS情報]

1

# 【参考】(3)比較実行(1/2)

## 定義した比較を実行する

比較定義詳細を設定したら、比較を実行します。

メニュー:比較 > 比較実行

① 比較実行で、下表のように選択または入力し、 [比較] ボタンを押下する。

空欄

② 比較結果が表示される。

IPアドレス 【OS情報-Gathered Facts】

比較実行					
比較定義: 3:IPアドレス 【100:OS情報 - 91:Gathered … 出力内容: <ul> <li>         ・全件出力         ・差分のみ         </li> <li>         比較     </li> </ul>	▼ 基準日1: (		基準日2:	対象ホスト:	選択
比較定義	基準日1	基準日2	出力内容		

空欄

全件出力

# 【参考】(3)比較実行(2/2)



比較項番	比較結果							
1       差分あり       targethost       OS情報       1       GatherFacts2       2021/10/01 09:25       192.0.2.2         2       差分あり       targethost       Gathered Facts       1       GatherFacts1       2021/07/28 16:47       192.0.2.1         Excel出力	比較項番⇔	結果⇔	ホスト名⇔	メニュー名称⇔	No⇔	オペレーション名令	基準日令	IPアドレス令
2 差分あり targethost Gathered Facts 1 GatherFacts1 2021/07/28 16:47 192.0.2.1 Excel出力 CSV出力	1	差分あり	targethost	OS ^r 情報	1	GatherFacts2	2021/10/01 09:25	192.0.2.2
Excel出力 CSV出力	2	差分あり	targethost	Gathered Facts	1	GatherFacts1	2021/07/28 16:47	192.0.2.1
	Excel出力 CSV出力							

# 3. 実習3【収集機能】ターゲットホストのSSL証明 書ファイルを収集する



## 実習3全体図

### 実習3の作業の流れ

●実習1と全体の流れは同じですが、収集対象がファイルになります。

パラメータシートから収集したファイルがダウンロードできるようになります。



# 3.1 ターゲットホストの登録

## ターゲットホストの接続情報を登録する

●実習1と同じホストを利用する場合は、この手順は不要です。

#### メニュー: 基本コンソール > 機器一覧

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。



			ログイン	ログィ	インパスワード	Ansible利用情報
HW 機器種別	ホスト名 (任意の名称)	IPアドレス (ご使用の環境 に応じて)	ユーザID (ご使用 の環境に	管理	ログイン パスワード (ご使用の環境	Legacy/Role利用 情報
			応じて)		に応じて)	認証方式
SV	targethost	192.0.2.1	root	•	*****	パスワード認証

## 3.2 オペレーションの登録

### 今回のオペレーションを登録する

今回のオペレーションを登録します。

### メニュー:基本コンソール > オペレーション一覧

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

		±≈।	宝炼买空口。	±∗		アクセス権
	NO. 1/10-23/10	オペレーション石	天肥了足口	ন	設定	アクセス許可ロール
	8 8	getSSL1	2021/04/23 17	:10	設定	
l						
	オペレーション名	実施予定 (は音のB	時間			
	(江志り石小)		2(B) <i>)</i>			
	getSSL1	2021/04/23	8 17:10			

## Ansible-LegacyのMovementを登録する

この後Playbookに紐付けて、SSL証明書を取得してくるジョブとなります。

### メニュー: Ansible-Legacy > Movement一覧

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

		海球カノマ		Ansible≉	川用情報
MOVEMENTID	Movement	産産タイマー	ホスト指定形式*	WinRM接続	ヘッダーセクション
自動入力	getSSL		IP v	<b>_</b>	

Movement名	Ansible利用情報
(任意の名称)	ホスト指定形式
getSSL	IP

## 作業実行用Playbookの作成(1/2)

- 「ターゲットホストの/tmp/直下に、SSL証明書ファイル取得のためのYAMLファイルを作成
   み
   作成したYAMLファイルをITAホストサーバの収集ディレクトリにコピーする
   →
   SSL証明書ファイルをITAホストサーバの収集ディレクトリにコピーする」という内容で
   す。
- ●ファイルの収集用ディレクトリについては、<u>『3.4.1 ファイルの収集用ディレクトリ』</u>を参照してください。
- 事前にターゲットホストの /etc/pki/tls/certs/ 直下に、SSL証明書ファイル(test.crt) を作成して配置してください。

## 3.4 Playbookの登録(2/3)

## 作業実行用Playbookの作成(2/2)

```
- name: make yaml file
blockinfile:
  create: yes
  mode: 0644
  insertbefore: EOF
  marker: ""
  dest: " /tmp/getSSL.yml"
  content: |
   SSL_file_name
                              : {{ VAR_ssl_name }}
   SSL_file
                          : {{ VAR_ssl_name }}
- name: copy the make yaml file to local
 fetch:
  src: "/tmp/getSSL.yml"
  dest: "{{___parameter_dir___}}/{{ inventory_hostname }}/"
  flat: yes
- name: get SSL file
fetch:
  src: "/etc/pki/tls/certs/{{ VAR_ssl_name }} "
  dest: "{{ __parameters_file_dir__ }}/{{ inventory_hostname }}/"
  flat: yes
```

#### ファイル名:getSSL.yml

## Ansible-LegacyにPlaybookを登録する

作成した作業実行用Playbookを登録します。

### メニュー: Ansible-Legacy > Playbook素材集

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

表材το	P1aybook麦材么*	Playbook素材		アクセス権		
TO TO THE	Figybook #ei/Jig			設定	アクセス許可ロール	
自動入力	getSSL	ファイルを選択	選択されていませ	ю	設定	
		事前アッ	プロード			
		アップロード状況 アップロードしま ファイル名 getSS サイズ568bytes	: した。 L.yml			

Playbook素材名 (任意の名称)	Playbook素材
getSSL	getSSL.yml

# 3.4.1 ファイルの収集用ディレクトリ(1/2)

## 収集したファイルは収集用のディレクトリに格納される

●収集したファイルは、ITAの予約変数で指定された収集用ディレクトリに格納します。

dest: "{{       parameters_file_dir			inventory_hostname }}/"	getSSL.yml 下から2行目
パスの変数		2	のディレクトリが指定されて	いる
	ITA予約変数		変数指定内容	
 ソースファイルの 格納先	parameter_dir		作業結果ディレクトリ配下の「_parameters」	のパス
収集したファイルの 格納先	parameters_file_dir_	_	作業結果ディレクトリ配下の「_parameters_	_file」のパス

# 3.4.1 ファイルの収集用ディレクトリ(2/2)

### 収集用のディレクトリは、下記のようなファイルの階層構造になっています。

#### ファイルの階層構造



## MovementとPlaybookを紐付ける

登録したMovementとPlaybookを紐付けます。

### メニュー: Ansible-Legacy > Movement-Playbook紐付

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

紐付項番	Movement*	Playbook素材	インクルード順序	アクセス権 設定 アクセス許可ロール
自動入力	2:getSSL 🔻	getSSL 🔻	1	設定

Movement	Playbook素材	インクルード順序
getSSL	getSSL	1

### 具体値登録用メニューを作成する

ファイル名(test.crt)を登録するためのパラメータシートを作成します。

#### メニュー:メニュー作成 > メニュー定義・作成

- ① 各項目で下表のように選択または入力する。
- ② [作成] ボタンを押下する。

#### 1.基本情報

メニュー名(任意の名称)	作成対象	表示順序
SSL証明書名	パラメータシート(ホスト/オペレーションあり)	4

#### 2.対象メニューグループ

入力用	代入値自動登録用	参照用
入力用(デフォルト)	代入値自動登録用(デフォルト)	参照用(デフォルト)

#### 3.項目

項目名(任意の名称)	入力方式	最大バイト数(任意の値)
ファイル名	文字列(単一行)	128

## 3.6 ファイル名の登録(2/3)



## 3.6 ファイル名の登録(3/3)

## ファイル名を登録する

作成したパラメータシートにファイル名(test.crt)を登録します。

#### メニュー: 入力用 > SSL証明書名

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

No	+764	オペレーション	パラメータ
No ホスト名*		オペレーション*	ファイル名
1	targethost 🔹	2021/04/23 17:10_8:getSSL1 •	test.crt

ホフトタ	オペレーション	パラメータ
		ファイル名
targethost	getSSL1	test.crt

### 代入値自動登録設定を登録する

- ●収集するファイル名(具体値)を、Playbook内の変数に紐づけます。([ファイル名: test.crt] [変数名: VAR_ssl_name])
- ●ファイル名を別のパラメータシートに登録しておき(<u>『3.6 ファイル名の登録』</u>で登録 済み)、代入値自動登録設定でパラメータシートの項目名とPlaybook内の変数を紐づけ ておくと、変数の具体値が自動的に設定されます。

### メニュー: Ansible-Legacy > 代入値自動登録設定

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

					•
	パラメータシート(From)				
項番 メニューグループ	-ב_א	項目*	登録方式	* Movement*	Value変数
自動入力 	L証明書名 ▼	パラメータ/ファイル名 🔻	Value型 🔻	2:getSSL 🔻	¥SXA 1:VAR_ssl_name ▼
パラメータシー	ト (From)			IaC変	数(To)
メニューグループ: 西日		登録フ	登録方式	Movement	Value変数
メニュー				Hovement	変数名
代入値自動登録用: SSI 証明書名	パラメータ/ ファイル名	Value	e型	getSSL	VAR_ssl_name

## 3.8 収集値を登録するパラメータシートの作成(1/3)

### 収集した値を登録するパラメータシートを作成する

- [SSL証明書] メニューを作成します。
- ●「ファイル名」「ファイル」の2つの項目を設置します。「ファイル」からは収集した ファイルがダウンロードできるようになります。

#### メニュー:メニュー作成 > メニュー定義・作成

① 各項目で下表のように選択または入力する。

② [作成] ボタンを押下する。

#### 1.基本情報

メニュー名(任意の名称)	作成対象	表示順序
SSL証明書	パラメータシート(ホスト/オペレーションあり)	2

#### 2.対象メニューグループ

入力用	代入値自動登録用	参照用
入力用(デフォルト)	代入値自動登録用(デフォルト)	参照用(デフォルト)

# 3.8 収集値を登録するパラメータシートの作成(2/3)

#### 3.項目







### 収集項目値管理を登録する

- ●収集した値をパラメータシートに自動登録するよう設定します。
- 収集項目(FROM)のYAMLファイル名・変数名と、パラメータシート(TO)のメニュー名・項目名を紐付けます。「ファイル名」「ファイル」の2つ分登録します。

### メニュー: Ansible共通 > 収集項目値管理

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

### 収集項目 (FROM)

パラメータシート (TO)

		収集項目(	FROM)		パラメータシート(T0)		
ID	パース形式*	PREFIX(ファイル名)	変数名*	メンパ変数	メニューグループ:メニュー*	項目*	
自動入力	YAML 🔻	getSSL	SSL_file_name		2100011611:代入値自動登録用:5:SSL証明書 ▼	パラメータ/ファイル名 🔻	

	収集項目(FR	ROM)	パラメータシート(TO)			
パース 形式	PREFIX (ファイル名)	変数名	メニューグループ :メニュー	項目		
YAML	getSSL	SSL_file_name	代入値自動登録用:SSL証明書	パラメータ/ファイル名		
YAML	getSSL	SSL_file	代入値自動登録用:SSL証明書	パラメータ/ファイル		

## 3.10 収集インターフェース情報の登録

### 収集インターフェース情報を登録する

- ●収集した値をITAのパラメータシートに登録する際のREST APIアクセスで必要になるため、RESTユーザー/パスワードを実行権限のあるユーザーで登録します。
- ●実習1と同じRESTユーザーを使用する場合は、この手順は不要です。

#### メニュー: Ansible共通 > 収集インターフェース情報

- ① [フィルタ]を押下する。
- 「一覧」に1行だけ表示されるので、 [更新] ボタンを押して下表のように入力し、
   [登録] ボタンを押下する。

一覧								
履歴 更新 ID⇔ ホスト名⇔ IP⇔ RES	STユーザー⊖ RESTパスワード inistrator ********	REST方式令 IP H	プロト⊐ル⇔ : http	ポート会 80				
10 #XFA IP*	administrator	•• 0	REST,	力式*	RESTユーザ	<u>_</u>	RES	5Tパスワード
				実行	5権限のあるユ	ーザー	そのユー	ザーのパスワード

# 3.11 作業実行(1/2)



Movementとオペレーションを選択し、作業実行します。

### メニュー: Ansible-Legacy > 作業実行

- ① Movement[一覧] から登録したMovementを選択する。
- ② オペレーション[一覧]から登録したオペレーションを選択する。
- ③ [実行] ボタンを押下する。

Movement[一覧]			۷
選択 MovementID _会 Movement名会 オーケストレータ会 選延タイマー会 ホスト指定形式会 MinBM接続会	nsible利用情報 Ansible Core利用情報 ヘッダーセクション会 オブションパラメータ会 virtualenx会	nsible.cfg アクセス様 アクセス許可ロール _会 帰考会 最終更新日時会	最終更新者会
6 GatherFacts Ansible Legacy IP - hosts: remote gather	all _user: "{{loginuser }}" _facts: yes	2022/05/19 16:49:54	データボータビリティブロシージャ
getSSL Ansible Legacy IP		2022/05/23 10:44:29	データボータビリティブロシージャ
フィルタ私果件数: 2			
オペレーション[フィルタ]			
オペレーション[一覧]			2
選択 No. ム オペレーションIDム オペレーション名会 実施予定日時会 最終実行日時会 アクセス権	编考入 影终史新日時人 影终史新者人		
アクセス許可ロール会 5 5 5 GatherFacts1 2021/04/22 17:09 2022/05/24 09:30	2022/05/24 09:30:38 legacy作業実行プロシージャ		
6 GatherFacts2 2021/10/01 09:25	2022/05/20 13:13:34 データボータビリティブロシージャ		
7 7 getSSL1 2021/04/23 17:10 2022/05/23 17:23	2022/05/23 17:23:11 legacy作業実行プロシージャ		
8 8 getSSL2 2021/04/28 12:19 2022/05/23 11:43	2022/05/23 11:43:23 legacy作業実行プロシージャ		
フィルタ結果件数: 4			
		Movement[一暫]	オペレーション
MovementID 7			
MovementID 7 Movement®, getSSL			
MovementID 7 Movement® getSL		getSSL	getSSL

# 3.11 作業実行(2/2)

## 作業のステータスを確認する

[作業状態確認] メニューで、ステータスが完了になれば、作業完了です。

### メニュー: Ansible-Legacy > 作業状態確認

Menu										
ンメニュー										
amont_F		項目		値						
iment—見	作業No.	作業No. 23								
pook素材集	実行種別			通常						
	ステータス	ステータス 完了								
vement-Playbook紐付	実行エンジン			Ansible Core						
物名一管	呼出元Symphony									
	呼出元Conducto	r								
值自動登録設定	実行ユーザ			システム管理者						
		ID	7							
対象ホスト		名称	getSSL							
、值管理		遅延タイマ(分)								
		Ansible利用情報	ホスト指定形式	IP						
実行	Movement	1115 10 10 10 10 10 10 10 K	WinRM接続							
		Ansible Core利用情報	virtualenv							
状態確認		Ansible Tower利用情報	virtualenv							
管理		Ansible Automation Controller利用情報	実行環境							
		ansible.cfg								
		No.		7						
	オペレーション	名称	getSSL1							

## 収集状況を確認する

収集の成功/失敗を確認します。

- メニュー: Ansible-Legacy > 作業管理
- ① [フィルタ] ボタンを押下する。
- ② 一覧 > 収集状況 > ステータス に、以下のように表示される。
  - · 対象外 : 収集失敗
  - 収集済み : 収集成功
  - 収集済み(通知あり):登録/更新中に不備があった場合
  - 収集エラー

: 登録したオペレーションかターゲットホストの情報に不備が あった場合



### パラメータシートを確認する

収集した値がパラメータシートに登録され、ファイルがダウンロードできるか確認しま す。

#### メニュー: 入力用(もしくは代入値自動登録用) > SSL証明書

① [フィルタ] ボタンを押下する。

② 一覧を表示し、作成した項目に値が入っているか確認する。

屈田	尼田 指制 百姓 南山	亩☆	また	また	市転	南山	扇正	扇正	廢止	感止	iosi⊧-	扇正	威止	廠止	感止	感止						<b>ナフトタム</b>			パラメ-	-タ
胆症 恆表	使表	史初	廃止 NO₹	「廃止	廃止	L NO	NO	NOŻ	▽ ハレヘトǎ▽	ID⇔	オペレーション名令	基準日時令	実施予定日時⇔	最終実行日時令	ファイル名令	ファイル										
履歴	複製	更新	廃止	1	targethost	8	getSSL1	2021/07/28 11:25	2021/04/23 17:10	2021/07/28 11:25	test.crt	<u>test.crt</u>														

# 4. 実習4 【比較機能】実習3で収集したSSL証明書 ファイルを、異なる日時に収集したファイルと比較する


## 実習4全体図(1/2)

## 実習4の作業の流れ

●実習3で収集したSSL証明書ファイルとは内容と収集日時が異なるSSL証明書ファイルを 収集し、両ファイルを比較します。





●「同一メニューだが収集した日時の異なる値」を比較する際は、比較実行の際に「基準日」を設定します。



全体図(比較)

## 4.1 オペレーションの登録

### オペレーションを登録する

比較用のオペレーションを登録します。

#### メニュー:基本コンソール > オペレーション一覧

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

No. オペレーション	+41 2-2.00	オペレーション名*	宝板又今日味。	アクセス権		
	オペレーション10		夫.肥了正口时**	設定	アクセス許可ロール	
自動入力	自動入力	get SSL2	2021/04/28 12:19	設定		

オペレーション名	実施予定時間			
(任意の名称)	(任意の時間)			
getSSL2	2021/04/28 12:19			

## 差分有りSSL証明書を用意しておく

- ●この実習では「差分あり」の表示を確認したいので、実習3で収集したファイルと差分のあるファイルを収集します。
- ターゲットサーバの/etc/pki/tls/certs/に入ってSSL証明書(test.crt)の中身を一部書 き換えておきます。



## ファイル名を登録する

●実習3で作成した[SSL証明書名]メニューに比較用のレコードを登録します。
 ●オペレーションのみ比較用にし、そのほかは実習3と同じ内容です。

#### メニュー: 入力用 > SSL証明書名

① 登録 > 登録開始を押下する。

② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

No	ホスト名*	オペレーション	パラメータ
		オペレーション*	ファイル名
2	targethost 🔹	2021/04/28 12:19_9:getSSL2	test.crt

ホフトタ	オペレーション	パラメータ		
		ファイル名		
targethost	getSSL2	test.crt		

## 4.4 作業実行

### 作業実行する

- ●差分有りのSSL証明書を収集します。
- Movementは実習3と同じくgetSSLで、オペレーションだけ比較用のgetSSL2にします。

#### メニュー: Ansible-Legacy > 作業実行

- ① Movement[一覧] から登録したMovementを選択する。
- ② オペレーション[一覧]から登録したオペレーションを選択する。
- ③ [実行] ボタンを押下する。

Movement[一覧]						۵		
選択 MovementID _会 Movement名会 オーケスト	レータ _会 遅延タイマー _会 ホスト指定形式 _会 WinRM接続会	Ansible利用情報 ヘッダーセクション _会 オプションパラメータ _会	Ansible Core利用情報 virtualenv会 ansible.cfg ア	アクセス権 備考 クセス許可ロール会	⇔ 最終更新日時会	最終更新者会		
6 GatherFacts Ansible Le	gacy IP	<pre>- hosts: all remote_user: "{(loginuser })" gather_facts: yes become: yes</pre>			2022/05/19 16:49:54	データボータビリティブロシージャ		
7 getSSL Ansible Le	jacy IP				2022/05/23 10:44:29	データボータビリティブロシージャ		
フィルタ結果件数:2								
オペレーション[ノイルタ]								
オペレーション[一覧]						Δ		
選択 No. 🕁 オペレーションID 👌 オペレーション	(名音) 実施予定日時音 最終実行日時音 アクセス アクセス許可に アクセス許可に	推 四一儿会 備考会 最終更新日時会 最終更新者会						
5 5 GatherFacts1	2021/04/22 17:09 2022/05/24 09:30	2022/05/24 09:30:38 legacy作業実行プロシージャ 2022/05/20 13:13:34 データボータドリティブロシー	- 05 87					
C 7 7 getSSL1	2021/04/23 17:10 2022/05/24 09:46	2022/05/24 09:46:12 legacy作業実行プロシージャ	57					
8 8 getSSL2	2021/04/28 12:19 2022/05/23 11:43	2022/05/23 11:43:23 legacy作業実行プロシージャ						
フィルタ結果件数:4								
					Mover	nent[一暫]	オペレー	ション「一覧
					HOVCI			
MovementID 7 Movement% getSSI								
				_	C	ietSSI	ae	tSSL2
					9	,0.00-	90	
ドライラン 実	Ŧ							

### パラメータシートを確認する

●getSSL2がパラメータシートに収集されているか確認します。

●<u>4.7 比較実行</u>で必要になるので、基準日時を確認しておきます。

#### メニュー: 入力用(もしくは参照用) > SSL証明書

① [フィルタ] ボタンを押下する。

② 一覧を表示し、作成した項目に値が入っているか確認する。

一覧	/更新													
屋田	屋田 塩制 古蛇 廃止 … ム ナマレクム				<u>جد جاملا ہے</u>					パラメータ				
KELDE	牧薮	史初	廃止	NOA	小∧∟₫⊅	ID⇔	オペレーション名令	基準日時	⇔	実施予定日	時令	最終実行日時⇔	ファイル名令	ファイル
履歴	複製	更新	廃止	1	targethost	8	getSSL1	2021/07/28	11:25	2021/04/23	17:10	2021/07/28 11:25	test.crt	<u>test.crt</u>
履歴	複製	更新	廃止	2	targethost	9	getSSL2	2021/07/28	12:08	2021/04/28	12:19	2021/07/28 12:08	test.crt	<u>test.crt</u>
4														
フィル	夕結果	件数:2												

### 比較対象となる2つのメニューを選択する

「同一メニューだが収集した日時の異なる値」を比較するので、比較対象メニューは同じ ものを選択します。

#### メニュー:比較 > 比較定義

- ① 登録 > 登録開始を押下する。
- ② 各項目で下表のように選択または入力し、 [登録] ボタンを押下する。

No	比較定義名*	比較対象メニュー1*	比較対象メニュー2*	全件一致
自動入力	SSL証明書	2100011611:代入値自動登録用:5:SSL証明書 ▼	2100011611:代入値自動登録用:5:SSL証明書 ▼	• •

比較定義名 (任意の名称)	比較対象メニュー1	比較対象メニュー2	全件一致
SSL証明書	代入值自動登録用:SSL証明書	代入值自動登録用:SSL証明書	•

## 4.7 比較実行(1/3)

### 定義した比較を実行する

●登録した比較定義「SSL証明書」を選択し、基準日を入力します。

- ●基準日は、SSL1と2それぞれが最新となる日時を設定します。
- ●基準日については次ページを参照してください。

#### メニュー:比較 > 比較実行

① 比較実行で、下表のように選択または入力し、 [比較] ボタンを押下する。

② 比較結果が表示される。

比較実行				
	エ四寺 【 217,001ぎエ四寺 217,001ぎエ	_ 其進口1 · 2021/7/28 11-30	11年日2 ・ 12021 /7 /28 12・20	
出力内容: ① 刍	正明音 (217.55に証明音 - 217.55に証・ 全件出力 () 差分のみ	₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩	₩₩¥₩₩¥₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩₩	
比較				
比較定義	基準日1	基準日2	出力内容	
SSL証明書	2021/7/28 11:30	2021/7/28 12:30	全件出力	

## 4.7 比較実行(2/3)

#### ●基準日の設定は以下のようになります。



収集の基準日に応じて比較の基準日を設定する。

## 4.7 比較実行(3/3)

# 比較結果が表示される

比較結果

比較項番⇔	結果⇔	ホスト名令	メニュー名称⇔	No⇔	オペレーション名令	基準日令	パラメータ/ファイル名令	パラメータ/ファイル令
1	差分あり	targethost	SSL証明書	1	getSSL1	2021/07/28 11:25	test.crt	test.crt
2	差分あり	targethost	SSL証明書	2	getSSL2	2021/07/28 12:08	test.crt	test.crt
Excel出力 CSV出力								
					2	圭分かめれ	は赤字で出力で	される。

