

Hitachi System Information Capture 基本ガイド

解説・手引・操作書

3021-3-731-10

前書き

■ 対象製品

P-9W65-3111 Hitachi System Information Capture 02-02-A (適用 OS : Red Hat Enterprise Linux Server 6 (64-bit x86_64))

P-2C65-3114 Hitachi System Information Capture 02-02-A (適用 OS : Windows Server 2008 R2 Standard (x64)、Windows Server 2012 R2 Standard、Windows 7 Professional (x64)、Windows 8.1 Pro(x64)、Windows 10 Pro(x64))

■ 輸出時の注意

本製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。なお、不明な場合は、弊社担当営業にお問い合わせください。

■ 商標類

HITACHI は、株式会社 日立製作所の商標または登録商標です。

Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。

Microsoft および Excel は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Red Hat は、米国およびその他の国で Red Hat, Inc. の登録商標もしくは商標です。

UNIX は、The Open Group の米国ならびに他の国における登録商標です。

Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows Server は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

その他記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

■ マイクロソフト製品のスクリーンショットの使用について

マイクロソフトの許可を得て使用しています。

■ 発行

2016 年 6 月 3021-3-731-10

■ 著作権

All Rights Reserved. Copyright (C) 2015, 2016, Hitachi, Ltd.

変更内容

変更内容 (3021-3-731-10) Hitachi System Information Capture 02-02-A

変更内容	変更箇所
適用 OS に Windows 10 Pro(x64)を追加した。	—

単なる誤字・脱字などはお断りなく訂正しました。

はじめに

■ マニュアルの目的

このマニュアルは、Hitachi System Information Capture の機能と使い方について説明したものです。

■ 対象読者

このマニュアルは、Hitachi System Information Capture を利用して、システムまたはアプリケーションをテストする開発者が対象です。対象読者に必要な知識を次に示します。

- 開発するシステムの機能要件および非機能要件
- システムの開発環境および言語
- システムの周辺環境（DB、ネットワーク、ジョブ管理など）および標準仕様

■ 関連マニュアル

このマニュアルに関連するマニュアルの名称と表記を次の表に示します。

表記	正式名称	資料番号
<i>Hitachi System Information Capture</i> リファレンス	<i>Hitachi System Information Capture</i> リファレンス	3021-3-732
<i>Hitachi System Information Capture - Visualizer</i> 基本ガイド	<i>Hitachi System Information Capture - Visualizer</i> 基本ガイド	3021-3-733
<i>Hitachi System Information Capture - Visualizer</i> リファレンス	<i>Hitachi System Information Capture - Visualizer</i> リファレンス	3021-3-734

■ このマニュアルでの表記

このマニュアルでは、製品名を次のように表記しています。

表記	製品名
Excel	Microsoft® Excel®
Linux	Red Hat Enterprise Linux Server 6
Windows	Microsoft® Windows Server® 2008 R2 Standard 日本語版
	Microsoft® Windows Server® 2012 R2 Standard 日本語版

表記		製品名
Windows	Windows 7 Professional (x64)	Microsoft® Windows® 7 Professional 日本語版 (64 ビット版)
	Windows 8.1 Pro(x64)	Windows® 8.1 Pro 日本語版 (64 ビット版)
	Windows 10 Pro(x64)	Windows® 10 Pro 日本語版 (64 ビット版)
システム検証支援基盤		Hitachi System Information Capture

Windows のメニュー名の表記は Windows Server 2008 R2 を前提としています。

■ フォルダーとパスの表記

このマニュアルでは、Windows および Linux で共通の内容の場合、Windows の「フォルダー」を「ディレクトリー」と表記しています。また、「¥」を「/」と表記しています。Windows の場合、「ディレクトリー」を「フォルダー」に、「/」を「¥」に置き換えてお読みください。

■ 英略語

このマニュアルで使用している英略語を次に示します。

英略語	英字での表記
CSV	Comma Separated Value
FTP	File Transfer Protocol
HTTP	Hyper Text Transfer Protocol
NIC	Network Interface Card
UOC	User Own Coding
UTF-8	8-bit UCS Transformation Format

■ KB (キロバイト) などの単位表記について

1KB (キロバイト), 1MB (メガバイト), 1GB (ギガバイト), 1TB (テラバイト) はそれぞれ 1,024 バイト, 1,024² バイト, 1,024³ バイト, 1,024⁴ バイトです。

目次

前書き	2
変更内容	3
はじめに	4

1	システム検証支援基盤 (Hitachi System Information Capture) の概要	8
1.1	システム開発におけるシステム検証支援基盤の位置づけについて	9
1.2	システム検証支援基盤を使ってテストを実行するメリット	11
1.2.1	システム検証支援基盤を使って簡単にテストデータを作成できる仕組みについて	11
1.2.2	システム検証支援基盤によるテストの結果を簡単に確認できる仕組みについて	13
1.2.3	異なる種類のプロトコルが混在したテストのサポートについて	13
1.3	システム検証支援基盤がサポートするテストとは	15
1.4	システム検証支援基盤を使ったテストの流れ	18
1.5	システム検証支援基盤を使う場合に必要なテスト計画	19
2	テスト環境の構築	20
2.1	テスト環境構築の流れ	21
2.2	テスト環境の構築に必要な情報	22
2.3	システム検証支援基盤をインストールする (Linux の場合)	23
2.4	システム検証支援基盤をインストールする (Windows の場合)	24
2.5	システム検証支援基盤を使用するための OS 環境設定	26
2.5.1	カーネルパラメーターを設定する (Linux の場合)	26
2.5.2	環境変数を設定する	26
2.6	システムテスト支援定義ファイルの作成	28
2.6.1	システムテスト支援定義ファイルとは	28
2.6.2	システムテスト支援定義ファイルを生成するツールを準備する	28
2.6.3	システムテスト支援定義ファイルを作成する	29
2.6.4	システムテスト支援定義ファイルを格納する	31
2.7	本番環境のネットワークスイッチのミラーポートを設定する	33
3	テストの準備	34
3.1	テストに必要な情報	35
3.2	テストデータを作成する	36
4	テストの実行	38
4.1	リプレイ形式のテストを実行する	39

4.2	リプレイ形式テストの結果の確認	40
4.2.1	比較エラー情報ファイルを参照してテスト結果を確認する	40
4.2.2	比較エラー情報ファイルを保存する	41
4.2.3	テスト結果データ（応答電文）を確認する	42
4.2.4	テスト結果データ（応答電文）を保存する	43
5	システム検証支援基盤の詳細	44
5.1	システム検証支援基盤を使ってできることの一覧	45

索引	46
-----------	-----------

1

システム検証支援基盤（Hitachi System Information Capture）の概要

システム検証支援基盤の位置づけは、開発中のシステムをテストするためのシミュレーターです。この章では、システム検証支援基盤を使用してどのようなテストができるのか、また、どのようなメリットがあるのかを説明します。また、テストの流れとテスト計画の概要も説明します。

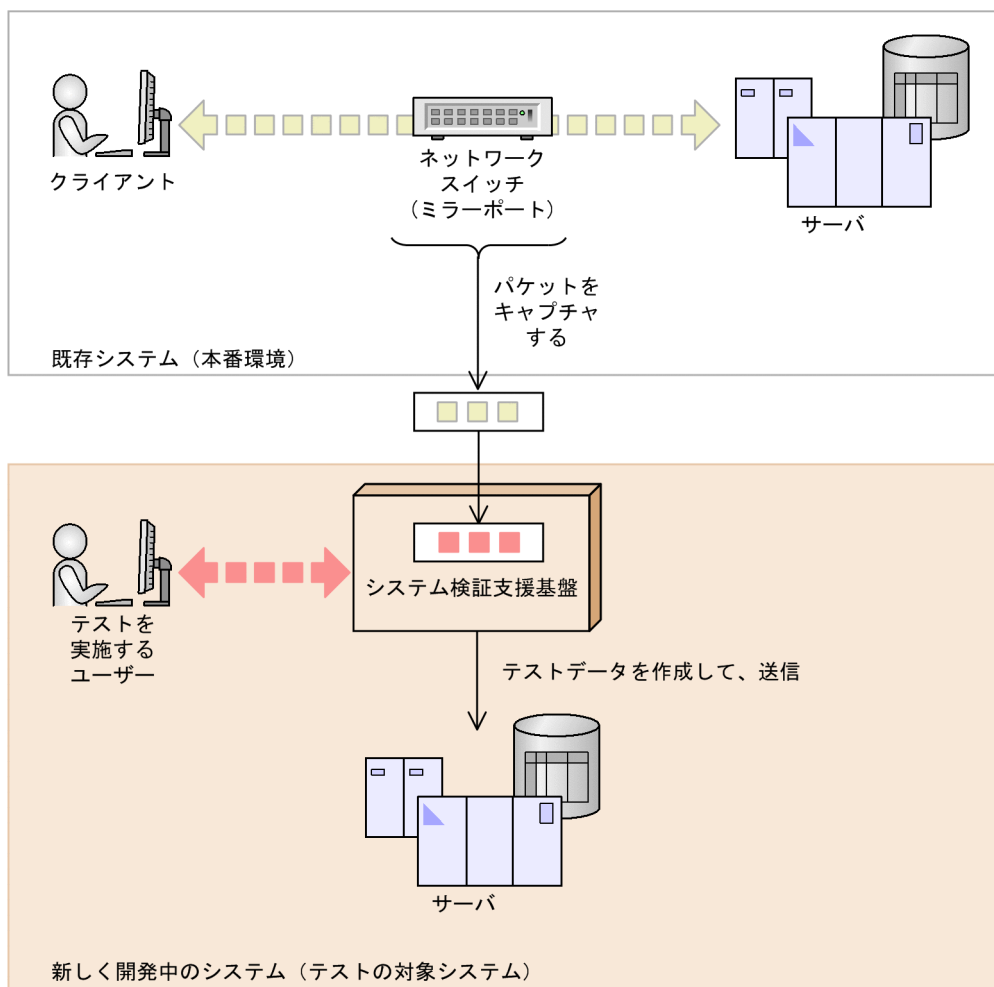
1.1 システム開発におけるシステム検証支援基盤の位置づけについて

システム開発に携わるユーザーは、開発中のシステムをテストするためのシミュレーターとしてシステム検証支援基盤を使用します。

既存の業務システムを新しい業務システムに入れ替える場合、新しく開発したシステムが計画時の要件のとおり機能することをあらかじめ確認するためにテストを実施します。また、新しいシステムが本番環境での運用に適しているかどうかを確認するためには、システム運用時に実際に処理するデータと同等のデータを使用してシミュレーションする必要があります。システムを開発するユーザーは、シミュレーターとしてシステム検証支援基盤を使用できます。

シミュレーターとしてのシステム検証支援基盤の位置づけを次の図に示します。

図 1-1 システム開発におけるシステム検証支援基盤の位置づけ



(凡例)

- : 本番環境のデータ
- : テストデータ

システム検証支援基盤は、稼働中の既存システムが処理しているデータのコピーを、パケット単位で取得（キャプチャ）します。キャプチャされたパケットは、システム検証支援基盤によってテスト対象のシステムに送られて、ユーザーはそのデータを使ってテストを実施します。

関連項目

- [1.3 システム検証支援基盤がサポートするテストとは](#)
-

1.2 システム検証支援基盤を使ってテストを実行するメリット

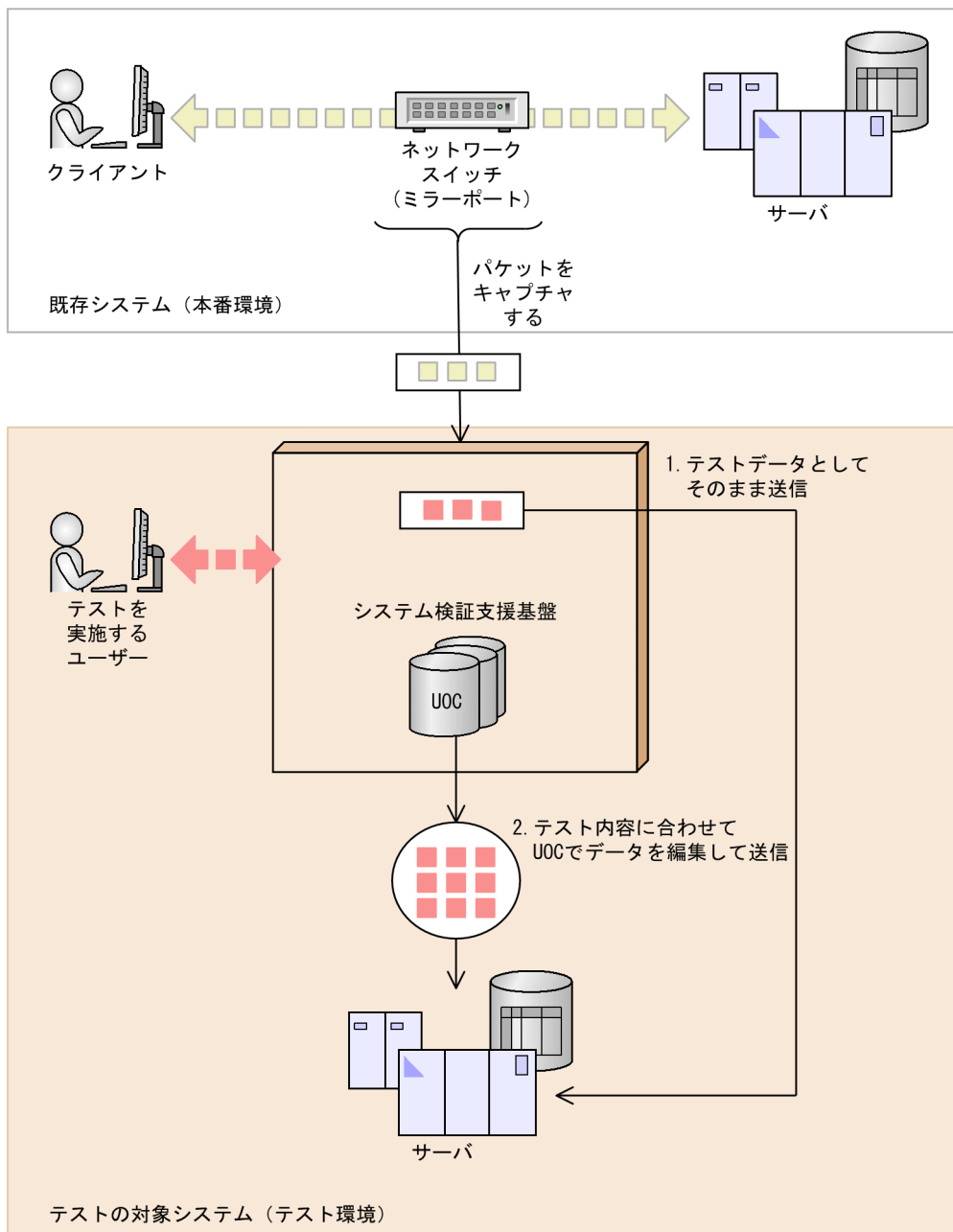
開発中のシステムをテストする場合に、システム検証支援基盤を使用するメリットを説明します。

1.2.1 システム検証支援基盤を使って簡単にテストデータを作成できる仕組みについて

システム検証支援基盤を使用すると、本番環境で稼働しているシステムが処理しているデータを、そのままテストデータとして使用できます。また、テストの内容に合わせた特殊なテストデータが必要な場合、テストを実施するユーザーは、本番環境で稼働しているシステムが処理しているデータと UOC (User Own Coding) を組み合わせることで、ゼロからテストデータを作る場合と比べて簡単にテストデータを作成できます。

システム検証支援基盤を使ってテストデータを作成する仕組みとメリットを次の図で説明します。

図 1-2 システム検証支援基盤を使ってテストデータを作成する仕組みとメリット



(凡例)

- : 本番環境のデータ
- : テストデータ

1. システムのテストの準備作業のうち、一般的に最も時間の掛かる作業の1つが、テストデータの作成です。テストデータは、テストのときだけ使用するダミーのデータですが、システムを正しくテストするためには、ある程度の妥当性が必要です。テストデータと本番環境のシステムが処理するデータが似ているほど、テストの信頼性は高くなります。

システム検証支援基盤は、本番環境のシステムが処理しているデータのコピーをキャプチャします。テストを実施するユーザーは、システム検証支援基盤がキャプチャしたデータをテストデータとしてそのまま使用できるため、短い準備時間で信頼性の高いテストを実施できます。

2. テストの内容によっては、本番環境のシステムが処理しているデータをそのまま使えないこともあります。例えば、特定のタイミングのデータや、エラー情報を含むデータが必要なテストもあります。

このような場合は、テストを実施するユーザーは、テストの内容に適したデータに書き換えるための処理を、あらかじめ UOC として作成しておきます。システム検証支援基盤の機能と UOC を組み合わせることによって、システム検証支援基盤がデータをキャプチャするタイミングでデータを選別したり、キャプチャしたデータがテストの対象システムに受け渡されるまでの間に、データを任意に編集したりできます。テストの内容ごとに UOC を作成しておけば、テストを実施するユーザーが毎回テストデータを作成する必要はありません。

1.2.2 システム検証支援基盤によるテストの結果を簡単に確認できる仕組みについて

システム検証支援基盤を使ってテストを実行すると、本番環境の既存システムからのレスポンスデータ（応答電文）とテストの対象システムからのレスポンスデータとが自動的に比較されます。比較結果は、比較エラー情報ファイルにまとめて出力されるため、テストの結果を簡単に確認できます。

テストで使用するプロトコルが RPC または RAP の場合は、比較エラー情報ファイルが生成されます。比較エラー情報ファイルに出力されるのは、テスト中にエラーが発生した処理、および本番環境の既存システムからのレスポンスデータとテストの対象システムからのレスポンスデータが不一致だった処理だけです。

システム検証支援基盤によるレスポンスの比較処理には、次の 2 種類があります。

- プロトコルレベルの比較

通常、システム検証支援基盤はプロトコルレベルでレスポンスデータを比較します。比較の対象は、RPC または RAP のレスポンスデータに含まれる戻り値です。

- データ内容の比較

レスポンスデータの内容を比較するためには、データ比較用の UOC (User Own Coding) が必要です。データ比較用の UOC を使ってテストを実施すると、処理結果が正常な（エラーが発生していない）レスポンスデータの内容だけを比較できます。

1.2.3 異なる種類のプロトコルが混在したテストのサポートについて

システム検証支援基盤は、複数の異なる種類のプロトコル (RPC、TCP など) を使用したテストに適用できます。

システム検証支援基盤がサポートしているプロトコルを次に示します。

- RPC
- RAP
- TCP

- UDP
- HTTP

システム検証支援基盤を使用すると、異なる種類のプロトコルが混在した状態のテストを実行できます。テストを実施するユーザーは、プロトコルごとにテストデータを作り分ける必要はありません。

関連項目

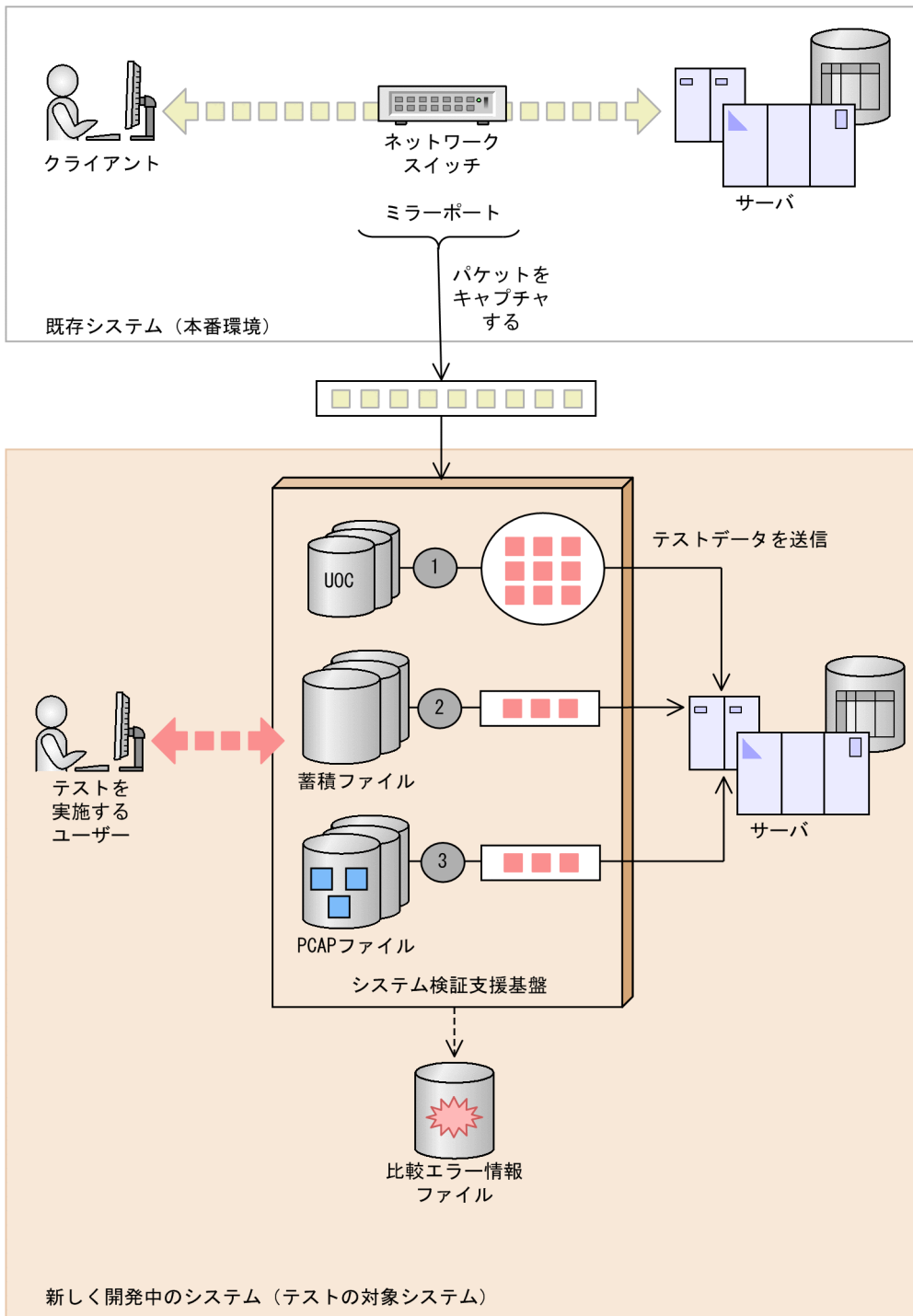
- [1.3 システム検証支援基盤がサポートするテストとは](#)
-

1.3 システム検証支援基盤がサポートするテストとは

システム検証支援基盤は、システムテストまたは統合テストのときに使用するのに適しています。システムテストまたは統合テストの内容は、例えばリグレッションテスト、ヒートラン、性能テストです。

システム検証支援基盤を使ったテストの形式の種類を次に示します。

図 1-3 システム検証支援基盤がサポートするテストの形式



(凡例)

■ : 本番環境のデータ

■ : テストデータ

1. 並行実行

既存システムからキャプチャしたデータを、その場でプロトコルに合ったデータ形式に作り変えて、テストの対象システムに送信する形式のテストです。

2. リプレイ実行

既存システムからキャプチャしたデータを、いったんファイルに蓄積しておき、蓄積ファイルをテストデータとして使用する形式のテストです。

3. PCAP 実行

既存システムのデータを PCAP 形式のファイルとしてキャプチャして、テストデータとして使用する形式のテストです。

既存システムからのレスポンスデータとテストの対象システムからのレスポンスデータが異なる処理の情報は、比較エラー情報ファイルにまとめて出力されます。

参考

リプレイ実行（リプレイ形式のテスト）を実施するためには、事前に蓄積実行という処理を実行して、テストデータを作成しておく必要があります。

1.4 システム検証支援基盤を使ったテストの流れ

一般的なシステムテストと同様に、システム検証支援基盤を使ったテストも事前にテスト計画が必要です。テスト計画が完成したら、テスト環境を構築します。テスト環境の構築完了後に、テストデータを準備します。

表 1-1 テストの流れ

順序	やること
1	テスト計画を立てる
2	テスト環境を構築する
3	テストデータを作成する
4	テストを実行する

関連項目

- 1.5 システム検証支援基盤を使う場合に必要なテスト計画
 - 2.1 テスト環境構築の流れ
 - 3.2 テストデータを作成する
 - 4.1 リプレイ形式のテストを実行する
-

1.5 システム検証支援基盤を使う場合に必要なテスト計画

システム検証支援基盤を使用してテストを実施する場合は、一般的なテスト計画に加えて、本番環境の既存システムで使用しているネットワークスイッチの仕様も確認しておく必要があります。また、UOC (User Own Coding) を活用してテストデータを加工するかどうかについても、テスト計画時に検討します。

システム検証支援基盤を使用したテストを実行するためには、テスト計画時に必ず次の項目を検討してください。

- 本番環境の既存システムのネットワークスイッチに、ミラーポートの設定をできるかどうか
システム検証支援基盤を使用する場合は、ミラーポートの設定を推奨します。ミラーポートを設定しない場合もシステム検証支援基盤を使用できますが、テストの形式はPCAP実行に限定されます。
- UOCが必要かどうか
例えば、本番環境の既存システムで処理されているデータに個人情報が含まれている場合は、本番環境のデータのコピーを、そのままテストデータとして使用できないこともあります。そのような場合は、本番環境のデータのコピーをテスト用に加工するためにUOCが必要です。テスト開始前までにUOCの作成が完了するように、スケジュールを検討してください。

2

テスト環境の構築

テスト環境は、システム検証支援基盤をインストールするマシンに構築します。インストール後は、OSの環境設定およびシステムテスト支援定義ファイルの作成が必要です。

2.1 テスト環境構築の流れ

テスト環境のマシンにシステム検証支援基盤をインストールしてください。インストールが完了したら、システムテスト支援定義ファイルを作成します。

表 2-1 テスト環境構築の流れ

順序	やること
1	構築に必要な情報を収集する（例：サーバの IP アドレス、ポート番号）
2	テスト用のマシンにシステム検証支援基盤をインストールする
3	システムテスト支援定義ファイルを作成する
4	本番環境の既存システムのネットワークスイッチにミラーポートの設定をする

関連項目

- [2.2 テスト環境の構築に必要な情報](#)
 - [2.3 システム検証支援基盤をインストールする（Linux の場合）](#)
 - [2.4 システム検証支援基盤をインストールする（Windows の場合）](#)
 - [2.5 システム検証支援基盤を使用するための OS 環境設定](#)
 - [2.6 システムテスト支援定義ファイルの作成](#)
 - [2.7 本番環境のネットワークスイッチのミラーポートを設定する](#)
-

2.2 テスト環境の構築に必要な情報

テスト環境をスムーズに構築するために、あらかじめ本番環境の既存システムのネットワーク構成やサーバ情報を確認しておくことを推奨します。

テスト環境を構築する前に、次に示す情報を収集してください。

- 本番環境の既存システムのネットワーク構成
- 本番環境の既存システムを構成している各種サーバの設定情報
各サーバの IP アドレスとポート番号を確認してください。
- テスト環境用のマシンに NIC (Network Interface Card) が 2 つ以上あること (並行実行の場合)
入力用 (パケットキャプチャ用) と出力用 (テストデータ送信用) に少なくとも 2 つの NIC が必要です。

2.3 システム検証支援基盤をインストールする (Linux の場合)

システム検証支援基盤は、テスト用のマシンにインストールしてください。システム検証支援基盤のインストールには、インストール用の CD-ROM を使用します。

前提条件

- スーパーユーザー権限でログインしている

操作手順

1. インストール用の CD-ROM を挿入します。

インストール用の CD-ROM の使い方の詳細は、リリースノートを参照してください。

インストールが正常に完了すると、次に示すディレクトリーが作成されます。このディレクトリーをシステム検証支援基盤のインストールディレクトリーと呼びます。

`/opt/HSIC`

2.4 システム検証支援基盤をインストールする (Windows の場合)

システム検証支援基盤は、テスト用のマシンにインストールしてください。システムテスト検証支援基盤のインストールには、インストール用の CD-ROM を使用します。

前提条件

- Administrators 権限でログインしている

操作手順

1. インストール用の CD-ROM を CD-ROM ドライブにセットします。

[日立総合インストーラ] ダイアログボックスが自動起動されます。

[日立総合インストーラ] ダイアログボックスが自動起動されない場合は、*CD-ROM* ドライブ:/HCD_INST.EXE を実行してください。

2. インストーラーの指示に従ってインストールを進めます。

3. [ユーザ情報] を指定します。

- [ユーザ名]
最大 50 文字の文字列を指定してください。
- [会社名]
最大 512 文字の文字列を指定してください。

4. [インストール先のフォルダ] でインストール先ディレクトリーを確認し、必要に応じてインストール先ディレクトリーを変更します。

- デフォルトでは、*OS*インストールドライブ:/Hitachi/HSIC/が設定されます。
- インストール先ディレクトリーを変更したい場合は、[変更] ボタンをクリックして任意のディレクトリーを選択してください。

選択した任意のディレクトリーに/Hitachi/HSIC/ディレクトリーが作成されて、そのディレクトリーにシステム検証支援基盤がインストールされます。

選択したディレクトリーが「~/Hitachi/」の場合は、ディレクトリー直下に/HSIC/ディレクトリーが作成されて、そのディレクトリーにシステム検証支援基盤がインストールされます。

選択したディレクトリーが「~/Hitachi/HSIC/」の場合は、そのディレクトリーにシステム検証支援基盤がインストールされます。

- ここで指定したインストール先ディレクトリーは、Windows のシステム環境変数HSIC_ROOT に設定されます。
- 絶対パスで 100 バイト以内の文字列を指定してください。
- UNC 表記はできません。
- ネットワークドライブは指定できません。

- インストール先ディレクトリーのパス中には、「#」およびマルチバイト文字を指定できません。
- 先頭が小文字の「u」で始まるディレクトリーは指定できません。

5. [プログラムをインストールする準備ができました] に表示された内容に問題がなければ、[インストール] ボタンをクリックします。

インストールの実行中に [キャンセル] ボタンをクリックしてインストールを中断した場合、システム検証支援基盤の実行環境が正常に作成されないおそれがあります。インストールを中断した場合は、再度、手順 1.から実行してください。

6. インストール処理が完了したら、マシンを再起動します。

システム検証支援基盤のインストールが完了します。

7. 次のどれかの条件に該当する場合は、インストール先ディレクトリーにアクセス権を追加するために、OS のicacls コマンドを実行します。

- Administrator 以外のユーザーがシステム検証支援基盤を使用する。
- OS のユーザーアカウント制御 (UAC) 機能が動作するパス (例 : C:/Program Files、C:/Program Files (x86)、C:/ProgramData、C:/Windows) にシステム検証支援基盤をインストールした。
- OS のユーザーアカウント制御 (UAC) 機能の「管理者承認モードですべての管理者を実行する」が有効になっている。

ユーザーアカウント制御 (UAC) 機能の設定の確認方法については、OS のマニュアルを参照してください。

コマンド実行例

```
icacls %HSIC_ROOT% /t /grant everyone:(OI)(F)
```

インストール先ディレクトリーにアクセス権が追加されて、Administrator 以外のユーザーもシステム検証支援基盤を使用できるようになります。

2.5 システム検証支援基盤を使用するための OS 環境設定

システム検証支援基盤を使ってテストの準備を始める前に、テスト環境の OS の環境設定が必要です。

2.5.1 カーネルパラメーターを設定する (Linux の場合)

パケットロス (システム検証支援基盤がキャプチャしたパケットが、処理の途中で消失してしまう問題) の発生は、カーネルパラメーターを設定することによって防止できます。

前提条件

- スーパーユーザー権限でログインしている

操作手順

1. カーネルパラメーター `net.core.rmem_max` に設定されている値を確認します。
2. カーネルパラメーター `net.core.rmem_max` に 20971520 よりも小さい値が設定されていた場合は、20971520 以上の値に変更します。

関連項目

- 2.3 システム検証支援基盤をインストールする (Linux の場合)
-

2.5.2 環境変数を設定する

システム検証支援基盤のコマンドを実行する前に、Linux の場合は `PATH` 環境変数、`LANG` 環境変数、`LD_LIBRARY_PATH` 環境変数を、Windows の場合は `PATH` 環境変数および `HSIC_LANG` 環境変数を設定することを推奨します。環境変数にコマンドのパスを指定しておく、コマンド実行時に毎回コマンドのパスを指定する必要がありません。また、システム検証支援基盤のメッセージを、日本語で出力するか英語で出力するかについても、環境変数で指定します。

前提条件

- スーパーユーザー権限または Administrators 権限でログインしている

操作手順

1. `PATH` 環境変数にシステム検証支援基盤のインストールディレクトリー `/bin` を指定します。
コマンド名を指定して実行するだけで、システム検証支援基盤のコマンドを実行できるようになります。

2. Linux の場合は、LD_LIBRARY_PATH 環境変数にシステム検証支援基盤のインストールディレクトリー/lib を指定します。

3. Linux の場合は、LANG 環境変数に、次のどちらかの値を指定します。

- C
コマンド実行時に英語のメッセージが出力されます。
- ja_JP.UTF-8
コマンド実行時に日本語 (UTF-8 コード) のメッセージが出力されます。

LANG 環境変数の設定を省略した場合、および上記以外の値を指定した場合の結果は、C を指定したときと同じです。

4. Windows の場合は、HSIC_LANG 環境変数に、次のどちらかの値を指定します。

- C
コマンド実行時に英語のメッセージが出力されます。
- ja_JP.SJIS
コマンド実行時に日本語 (SJIS コード) のメッセージが出力されます。

HSIC_LANG 環境変数の設定を省略した場合、および上記以外の値を指定した場合の結果は、C を指定したときと同じです。

5. シェルのログインスクリプトにPATH 環境変数、LD_LIBRARY_PATH 環境変数、またはLANG 環境変数を指定している場合は、ログインスクリプトの指定値を変更します。

手順 1.~手順 3.で指定した値と同じ値を、ログインスクリプトにも指定してください。

関連項目

- 2.3 システム検証支援基盤をインストールする (Linux の場合)
 - 2.4 システム検証支援基盤をインストールする (Windows の場合)
-

2.6 システムテスト支援定義ファイルの作成

システムテスト支援定義ファイルは、専用のツールを使用すると簡単に作成できます。

2.6.1 システムテスト支援定義ファイルとは

システム検証支援基盤を使ってテストを実行するには、システムテスト支援定義ファイルが必要です。例えば、既存システムの IP アドレスやテストの対象システムのポート番号などは、テストを開始する前に、システムテスト支援定義ファイルに定義しておく必要があります。

システムテスト支援定義ファイルは、テスト識別子ごとに作成してください。システムテスト支援定義ファイルのファイル名および格納場所は、次のとおりです。

ファイル名

システムテスト支援定義ファイルの名称は、テスト識別子と同じ名称にする必要があります。

格納場所

システムテスト支援定義ファイルのデフォルトの格納場所は、次に示すディレクトリーです。

```
システム支援検証基盤のインストールディレクトリー/conf
```

デフォルト以外のディレクトリーにシステムテスト支援定義ファイルを格納した場合は、テスト実行時にコマンドでシステムテスト支援定義ファイルのディレクトリーパスを指定する必要があります。

2.6.2 システムテスト支援定義ファイルを生成するツールを準備する

システムテスト支援定義ファイルは、専用のツールを使用すると簡単に作成できます。システムテスト支援定義ファイルを生成するツール (hsic_def.xls) を使用するには、Excel が必要です。

背景

テキストエディターを使ってシステムテスト支援定義ファイルを作成する場合、あらかじめ記述規則や定義項目の名称を正確に理解しておく必要があります。一方、専用のツール (定義生成 Excel) を使用すると、表に必要事項を記入していく感覚で、簡単にシステムテスト支援定義ファイルを作成できます。

前提条件

- システム検証支援基盤のインストールが完了している
- システム検証支援基盤をインストールしたマシンとは別のマシンで、システムテスト支援定義ファイルを作成する (Linux の場合)

操作手順

1. システムテスト支援定義ファイルの作成作業を実施するマシンに`hsic_def.xls` ファイルを移動させます。システム検証基盤をインストールしたマシンの OS が Windows の場合は、同じマシンでシステムテスト支援定義ファイルを作成することもできます。その場合は、ファイルを移動する必要はありません。

ファイルが格納されている場所（移動元）

システム検証支援基盤のインストールディレクトリー/`tool/def`

ファイルを格納する場所（移動先）

任意です。

FTP を使ってファイルを移動する場合

バイナリ転送モードを指定してください。

関連項目

- [2.6.3 システムテスト支援定義ファイルを作成する](#)

2.6.3 システムテスト支援定義ファイルを作成する

システムテスト支援定義ファイルには、本番環境の既存システムおよびテストの対象システムの情報を定義します。専用のツール（定義生成 Excel）を使用すると、簡単にシステムテスト支援定義ファイルを作成できます。

前提条件

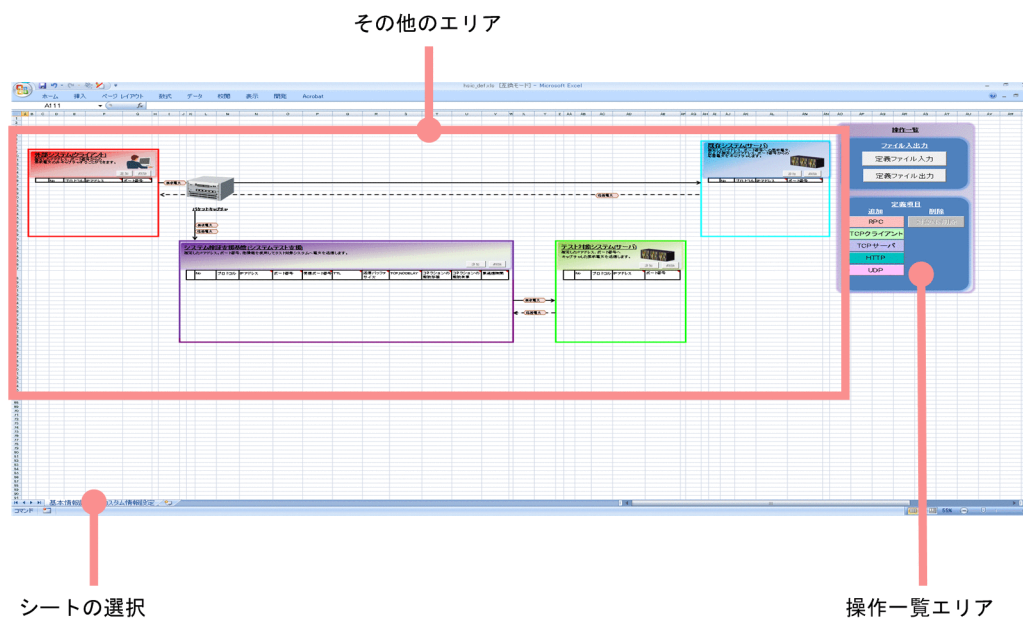
- システムテスト支援定義ファイルを生成するためのツール（`hsic_def.xls`）の準備が完了している
- Excel がインストールされている
- Excel のマクロの設定で、`hsic_def.xls` に含まれているマクロが有効になっている

参考

操作手順で説明する操作以外の操作（例：コピー&ペースト、元に戻す）は、実行しないでください。

操作手順

1. hsic_def.xls ファイルをダブルクリックします。



2. [基本情報設定] シートの右側にある [操作一覧] エリアで、テストで使用するプロトコルの種類に該当するボタンをクリックします。

使用するプロトコルが複数ある場合は、必要な回数だけ繰り返してボタンをクリックしてください。

[既存システム(サーバ)] エリア、[テスト対象システム(サーバ)] エリア、および [システム検証支援基盤(システムテスト支援)] エリアに行が追加されます。

3. 追加された行の、空白になっているセルを選択して、必要な値を入力または選択します。

入力規則、入力形式、およびデフォルト値について知りたい場合は、値を入力しようとしているセルを選択すると、説明が表示されます。また、入力内容の詳細を確認したい場合は、列の先頭の行のセルにカーソルを合わせると、説明が表示されます。

4. 本番環境の既存システムにクライアントが存在する場合は、[外部システム(クライアント)] エリアの [追加] ボタンをクリックします。

5. 追加された行の、空白になっているセルを選択して、必要な値を入力または選択します。

入力規則、入力形式、およびデフォルト値について知りたい場合は、値を入力しようとしているセルを選択すると、説明が表示されます。また、入力内容の詳細を確認したい場合は、列の先頭の行のセルにカーソルを合わせると、説明が表示されます。

6. [操作一覧] エリアの [定義ファイル出力] ボタンをクリックします。

省略できない項目の値を入力していなかったり、入力した値に誤りがあったりした場合は、警告メッセージが表示されます。[OK] ボタンをクリックすると、警告メッセージは消えて、問題があるセルが自動的に選択されます。セルの内容を確認し、修正してください。

すべての項目が正しく設定されている場合だけ、システムテスト支援定義ファイルを保存するディレクトリを選択するための画面が表示されます。

7. 任意のディレクトリーを選択して、[保存] ボタンをクリックします。

特に理由がない限り、デフォルトで表示されるファイル名をそのまま使用してください。
選択したディレクトリーにHSIC00 というファイルが保存されます。

8. [Office] ボタンから [Excel の終了] ボタンをクリックします。

ファイルを保存するかどうかを確認するメッセージが表示されます。

9. [はい] ボタンまたは [いいえ] ボタンをクリックします。

ツールが終了します。

関連項目

- 2.6.2 システムテスト支援定義ファイルを生成するツールを準備する
- 2.6.4 システムテスト支援定義ファイルを格納する

2.6.4 システムテスト支援定義ファイルを格納する

システムテスト支援定義ファイルは、システム検証支援基盤をインストールしたマシンに格納する必要があります。システムテスト支援定義ファイルを作成したマシンから、システム検証支援基盤をインストールしたマシンへ、マシン間でファイルを移動させてください。

前提条件

- システムテスト支援定義ファイルを作成済みである

操作手順

1. システムテスト支援定義ファイルを、システム検証支援基盤をインストールしたマシンに移動させます。

ファイルの移動手段は任意です。FTP を使ってファイルを移動する場合は、バイナリ転送モードを指定してください。

2. 移動させたシステムテスト支援定義ファイルを、次のディレクトリーに格納します。

システム検証支援基盤のインストールディレクトリー/conf

このディレクトリーにシステムテスト支援定義ファイルを格納した場合は、テストを実行するコマンドを実行するときに、システムテスト支援定義ファイルが格納されているディレクトリーのパスを指定する必要がありません。

参考

システム検証支援基盤のインストールディレクトリー/conf ではなく、システム検証支援基盤のインストールディレクトリー/job/テスト識別子以外の、任意のディレクトリーにシステムテスト支援定義ファイルを格納することもできます。任意のディレクトリーにシステムテスト支援

定義ファイルを格納した場合は、テストを実行するコマンドを実行するたびにシステムテスト支援定義ファイルが格納されているディレクトリーを指定する必要があります。

なお、テスト識別子とシステムテスト支援定義ファイルの名称は、同一です。

関連項目

- 2.6.3 システムテスト支援定義ファイルを作成する
-

2.7 本番環境のネットワークスイッチのミラーポートを設定する

システム検証支援基盤を使ってテストを実行するために、本番環境のシステムで使用しているネットワークスイッチのミラーポートの設定を有効にします。ミラーポートの設定手順については、ご使用のネットワークスイッチのマニュアルを参照してください。

3

テストの準備

テストを開始する前に、準備作業としてテスト計画書の確認およびテストデータの作成を実施します。

3.1 テストに必要な情報

システム検証支援基盤を使用したテストを実行する前に、いつテストを実施すればよいのか、また、どのようなテストを実施すればよいのかを確認します。

システム検証支援基盤を使用したテストを実行する前に、テスト計画書を参照して次の項目を確認してください。

- テストの内容
どのような特徴を持つデータが、テストデータとして必要なのかを確認します。
- テストスケジュール
テストを開始する時期、およびテストを完了する時期を確認します。
- 業務スケジュール
テストデータとして適切なデータを取得できるタイミングがいつなのかを確認します。

関連項目

- [3.2 テストデータを作成する](#)
-

3.2 テストデータを作成する

システム検証支援基盤の `esstart` コマンドのオプションで蓄積実行を指定して実行すると、テストデータを作成できます。

前提条件

- スーパーユーザー権限または Administrators 権限でログインしている
- PATH 環境変数が設定されている
- LD_LIBRARY_PATH 環境変数が設定されている (Linux の場合)
- システムテスト支援定義ファイルがシステム検証支援基盤のインストールディレクトリー/`conf` に格納されている
- テスト識別子が HSIC00 である

操作手順

1. テストデータを作成するために、`esstart` コマンドを実行します。

```
esstart -g HSIC00 -a
```

本番環境の既存システムでやり取りされているリクエストデータ (要求電文) とレスポンスデータ (応答電文) が、蓄積ファイルとして生成されます。これらの蓄積ファイルが、テストデータです。

例のとおりコマンドを実行した場合のファイル名およびファイルの生成場所を、次の表に示します。

表 3-1 蓄積ファイルの種類

データ種別	蓄積ファイルの名称と生成場所
本番環境のリクエストデータ	システム検証支援基盤のインストールディレクトリー/ <code>job/HSIC00/afile/req/HSIC00req00001</code>
本番環境のレスポンスデータ	システム検証支援基盤のインストールディレクトリー/ <code>job/HSIC00/afile/rpy/HSIC00rpy00001</code>

参考

システム検証支援基盤は、データ種別ごとに最大 10,000 ファイルの蓄積ファイルを生成できます。蓄積ファイルの名称の末尾には、連番で 00001 から 10000 までの番号が付きます。

2. テストデータの作成を終了するために、`esstop` コマンドを実行します。

なお、テストデータの作成の終了時刻をシステムテスト支援定義ファイルまたは `esstart` コマンドのオプションで指定している場合は、この手順は不要です。

```
esstop -g HSIC00
```

蓄積ファイルの生成処理が停止します。

3. 続けてほかのテスト用のテストデータを作成する場合は、その前に手順 1.と手順 2.で作成したテストデータをほかのディレクトリーに移動します。

参考

- テストデータ（蓄積ファイル）のファイル名は変更しないでください。
- テスト識別子が同じ場合、移動しなかったファイルは削除されます。

関連項目

- [3.1 テストに必要な情報](#)
-

4

テストの実行

この章では、リプレイ形式のテストの実行手順、およびテスト結果の確認手順を説明します。

4.1 リプレイ形式のテストを実行する

システム検証支援基盤のesstart コマンドを実行すると、リプレイ形式のテストを実行できます。

前提条件

- スーパーユーザー権限または Administrators 権限でログインしている
- PATH 環境変数が設定されている
- LD_LIBRARY_PATH 環境変数が設定されている (Linux の場合)
- テストデータ (蓄積ファイル) を作成済みである
- システムテスト支援定義ファイルがシステム検証支援基盤のインストールディレクトリー/conf に格納されている
- テスト識別子が HSIC00 である

操作手順

1. リプレイ形式のテストを実行するために、esstart コマンドを実行します。

```
esstart -g HSIC00 -r
```

リプレイ形式のテストの処理

テストデータのうち、テスト環境のリクエストデータ (本番環境のリクエストデータ (要求電文) から作成したテストデータ) が、テストの対象システムに送信されます。

プロトコルが RPC または RAP の場合は、テストの対象システムのレスポンスデータ (応答電文) と、本番環境のレスポンスデータとが自動的に比較されます。比較した結果、エラーや差異があった場合は、比較エラー情報ファイルが生成されます。

すべてのテストデータの処理が完了すると、リプレイ形式のテストは自動的に終了します。

関連項目

- [3.2 テストデータを作成する](#)
-

4.2 リプレイ形式テストの結果の確認

リプレイ形式のテストを実行すると、テスト結果として比較エラー情報ファイルが生成されます。比較エラー情報ファイルは CSV 形式なので、任意のツール（テキストエディターなど）を使って内容を確認できます。また、テストの対象システムからのレスポンスデータ（応答電文）も、テスト結果データとして保存されます。

4.2.1 比較エラー情報ファイルを参照してテスト結果を確認する

システム検証支援基盤を使ったリプレイ形式のテストが正常に終了すると、比較エラー情報ファイルが生成されます。比較エラー情報ファイルには、問題があった処理の情報だけがまとめられているため、このファイルを参照することによって、迅速にテスト結果を確認できます。

前提条件

- リプレイ形式のテストを実行済みである

参考

比較エラー情報ファイルは、テストを実行するたびに生成されます。比較エラー情報ファイルは、新しくテストを実行するときに自動的に前回のファイルに上書きされるため、手動での削除は不要です。

操作手順

- 次に示すディレクトリーを参照します。

```
システム検証支援基盤のインストールディレクトリー/job/HSIC00 (テスト識別子) /spool/dceinf/  
cmp
```

- ディレクトリー内にある比較エラー情報ファイル *HSIC00err001* (テスト識別子 + *err* + 通し番号) (CSV 形式) を開きます。

ファイルを開くためのツールは任意です。テキストエディターを使って比較エラー情報ファイルを開いた例を、次に示します。

```
P-9W65-3111 02-01    2015/02/05 15:44:22 980.387 HSIC00  
  
TIME  PROTOCOL TYPE  KEY  REQUEST RECEIVE TIME  REPLY RECEIVE TIME  REQUEST SEND  
TIME(TEST)  REPLY RECEIVE TIME(TEST)  ERROR TYPE  SRV-GROUP  SRV-SERVICE  RETURN CODE  
RETURN CODE(TEST)  RETURN CODE(UOC)  DETAIL INFORMATION  
  
2015/02/05 15:44:23 642.460  RPC  
5f52504300009b930e11d354a45e0d00660010ac5ba200000000000000000000  2015/02/05 15:44:23  
584.245  2015/02/05 15:44:23 584.264  2015/02/05 15:44:23 622.676  2015/02/05 15:44:23  
631.665  USER  sbdemo  srv01  0  0  29  zaiko-info(在庫情報)
```


上記は、テスト識別子が HSIC00、テストで使用したプロトコルが RPC の場合の出力例です。ERROR TYPE が USER であることから、既存システムのレスポンスデータ（応答電文）とテストの対象システムのレスポンスデータの内容が不一致だったということがわかります。日時は、それぞれリクエストデータ（要求電文）またはレスポンスデータを受信したタイミングを示しています。

不一致の原因の調査方法の例

- 比較エラー情報ファイルの RETURN CODE(UOC)の値（上記の例では 29）を参考にして、レスポンスデータのどの部分が不一致と判定されたのかを特定する。
- 比較エラー情報ファイルの KEY の値（上記の例では 5f52504300009b930e11d354a45e0d00660010ac5ba20000000000000000000000）を参考にして、どのリクエストデータまたはどのレスポンスデータに問題があったのかを特定する。
リクエストデータまたはレスポンスデータを特定できたら、リクエストデータと業務ロジックを突き合わせたり、レスポンスデータを比較したりする。

関連項目

- 4.1 リプレイ形式のテストを実行する
 - 4.2.2 比較エラー情報ファイルを保存する
-

4.2.2 比較エラー情報ファイルを保存する

比較エラー情報ファイルの 1 ファイル当たりの容量は、システムテスト支援定義ファイルに定義しておきます。1 つ目の比較エラー情報ファイルの容量が決められた値を超えると、2 つ目の比較エラー情報ファイルが生成されます。次の比較エラー情報ファイルの生成が始まるとき（KFSE80951-I メッセージが出力されるとき）に、古い比較エラー情報ファイルを保存することを推奨します。

背景

システム検証支援基盤が生成する比較エラー情報ファイルの数と容量は、ユーザーがシステムテスト支援定義ファイルに定義しておきます。比較エラー情報ファイルの数が、システムテスト支援定義ファイルに定義したファイル数の上限を超えると、最も古い比較エラー情報ファイルが上書きされます。

1 回のテストで複数の比較エラー情報ファイルが生成される場合は、生成された比較エラー情報ファイルをほかのディレクトリーに移動しておくことを推奨します。

操作手順

1. KFSE80951-I メッセージが出力されているかどうかを確認します。

KFSE80951-I メッセージが出力されていない場合は、以降の手順は不要です。

2. 次に示すディレクトリーを参照します。

システム検証支援基盤のインストールディレクトリー/job/テスト識別子/spool/dceinf/cmp

3. ディレクトリー内にある比較エラー情報ファイルのうち、最新のファイル以外を任意の別ディレクトリーに移動します。

比較エラー情報ファイルの新旧は、KFSE80951-I メッセージを参照することで判断できます。KFSE80951-I メッセージには、スワップ元とスワップ先のファイル名が出力されています。スワップ元よりもスワップ先の比較エラー情報ファイルの方が、新しいファイルです。

最新の比較エラー情報ファイルは、システム検証支援基盤によって更新中の可能性があるため、移動したり削除したりしないでください。

関連項目

- [4.2.1 比較エラー情報ファイルを参照してテスト結果を確認する](#)

4.2.3 テスト結果データ（応答電文）を確認する

システム検証支援基盤を使ったリプレイ形式のテストが正常に終了すると、テストの対象システムからのレスポンスが、応答電文ファイルとして生成されます。応答電文ファイルを参照することによって、詳細なテスト結果を確認できます。

前提条件

- リプレイ形式のテストを実行済みである

操作手順

1. 次に示すディレクトリーを参照します。

システム検証支援基盤のインストールディレクトリー/*job/HSIC00* (テスト識別子) /*afile/rpyts/*
このディレクトリーが存在しない場合は、システムテスト支援定義ファイルの*accumulate_file* の指定値 (テスト環境のテストデータを作成するかどうかの設定、およびディレクトリーのパス) を確認してください。

2. ディレクトリー内にある応答電文ファイルの内容を確認するために、*esafllfills* コマンドを実行します。

```
esafllfills -e r -s rpyts -g HSIC00 (テスト識別子) -p rpc システム検証支援基盤のインストールディレクトリー/job/HSIC00 (テスト識別子) /afile/rpyts/
```

応答電文ファイルの名称は、テスト識別子、応答電文であることを示す ID、および通し番号から構成されています。例えば、テスト識別子が *HSIC00* の場合のファイル名は、*HSIC00rpyts00001* です。ファイルが複数あるときは、ファイル名の末尾の通し番号によってファイルを区別します。

例のとおりコマンドを実行すると、指定したディレクトリー下に格納されているすべての応答電文ファイルの内容を確認できます。

関連項目

- 4.1 リプレイ形式のテストを実行する
 - 4.2.4 テスト結果データ（応答電文）を保存する
-

4.2.4 テスト結果データ（応答電文）を保存する

テストの対象システムからの応答電文を保存しておくこと、例えば、テストのエビデンスとして利用できます。

前提条件

- リプレイ形式のテストを実行済みである

操作手順

1. 次に示すディレクトリーを参照します。

システム検証支援基盤のインストールディレクトリー/*job/HSIC00*（テスト識別子）/*afire/rpyts/*
このディレクトリーが存在しない場合は、システムテスト支援定義ファイルの*accumulate_file*の指定値（テスト環境のテストデータを作成するかどうかの設定、およびディレクトリーのパス）を確認してください。

2. ディレクトリー内にある応答電文ファイル*HSIC00rpyts00001*（テスト識別子+*rpyts*+通し番号）を任意の別ディレクトリーに移動します。

ファイルを移動するときに、ファイルの名称は変更しないでください。ファイルの名称を変更すると、ファイルの内容を確認するためのコマンド（*esafllfills*）を実行したときに、正常に内容を確認できないおそれがあります。

関連項目

- 4.2.3 テスト結果データ（応答電文）を確認する
-

5

システム検証支援基盤の詳細

この章では、このマニュアルでは詳細を説明していない、システム検証支援基盤の応用的な機能を紹介します。また、それらの機能の目的と利用想定シーンも説明します。

5.1 システム検証支援基盤を使ってできることの一覧

このマニュアルで説明している機能以外にも、システム検証支援基盤には多種多様なテストを可能にしたり、テスト作業の効率向上をサポートしたりする機能があります。

システム検証支援基盤の機能のうち、このマニュアルでは詳細を説明していない機能を次に示します。これらの機能の詳細については、マニュアル「Hitachi System Information Capture リファレンス」を参照してください。

表 5-1 システム検証支援基盤の機能

やりたいこと (目的)	想定シーン	機能の名称
ミラーポートを使用しないで、テストデータを作成したい。	<ul style="list-style-type: none">テスト計画テスト実行	<ul style="list-style-type: none">PCAP 実行PCAP 形式ファイル
本番環境の既存システムが処理しているデータを、テストデータとしてリアルタイムに流用したい。	<ul style="list-style-type: none">リグレッションテスト長期間、連続してテストデータを流し続けて、問題が発生しないことを確認する必要があるとき	並行実行
すべてのクライアントからのリクエストデータ (要求電文) を、時系列に処理したい。	クライアント間でやり取りされるデータに依存するテスト	シリアルライズ処理
本番環境の既存システムからキャプチャしたデータの一部だけをテストデータとして使用したい。	<ul style="list-style-type: none">特定のクライアントおよびサーバのデータだけを使用したテスト特定の時間帯のデータだけを使用したテスト	フィルタリング
特殊なプロトコルを使用したい。	システム検証支援基盤がサポートしていないプロトコルのデータを使用したテスト	ユーザーテスト支援フレームワーク
本番環境の既存システムからキャプチャしたデータを編集してから、テストデータとして使いたい。	<ul style="list-style-type: none">異常値を含むデータを使ったテストキャプチャしたデータに個人情報が含まれているため、マスキングが必要キャプチャしたデータが暗号化されているため、復号処理が必要	<ul style="list-style-type: none">蓄積ファイル編集ユーティリティUOC (User Own Coding)
レスポンスデータ (応答電文) を比較するときに、ユーザーデータ領域を比較したい。	ユーザーデータの内容に依存するテスト	データ比較 UOC

索引

E

esstart コマンド 36, 39
esstop コマンド 36, 39

H

HSIC_LANG 環境変数 26
HTTP 13

L

LANG 環境変数 26
LD_LIBRARY_PATH 環境変数 26

P

PATH 環境変数 26
PCAP 実行 15

R

RAP 13
RPC 13

T

TCP 13

U

UDP 13
UOC 11
UOC(データ比較) 13

い

インストール
Linux の場合 23
Windows の場合 24

お

応答電文ファイル 42

か

カーネルパラメーター 26
環境変数 26

き

機能の詳細 45

け

計画 19, 35

こ

構築に必要な情報 22

し

システム検証支援基盤の位置づけ 9
システムテスト支援定義ファイル 28, 29
システムテスト支援定義ファイルの格納場所 31
準備 35

ち

蓄積実行 15
蓄積ファイル 36

て

定義生成 Excel 28, 29
テスト環境構築の流れ 21
テスト計画 19, 35
テスト結果データ 42
テスト結果を確認する仕組み 13
テスト識別子 36
テスト準備 35
テストデータを作成する 36
テストデータを作成する仕組み 11
テストの種類 15
テストの流れ 18

ね

ネットワークスイッチ 33

は

パケットロス 26

ひ

比較エラー情報ファイル 13, 40

ふ

プロトコルの種類 13

プロトコルレベルの比較 13

へ

並行実行 15

ほ

保存

 応答電文 43

 比較エラー情報ファイル 41

み

ミラーポート 33

よ

容量

 比較エラー情報ファイル 41

り

リプレイ形式 39

リプレイ形式テストの結果 40

リプレイ実行 15