

Jupyter Notebookの拡張機能を使って リモート機器にSSHして作業する

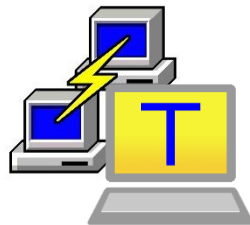
上野 優（株式会社富士通研究所）

2019.07.24
JANOG44ミーティング

■SSH (Tera TermやPutty) による機器の運用は今後も なくなる

■特にこんな場合

- 試行錯誤が必要
- トラブル対応
- 1度キリの作業



Jupyter Notebookでたのしく運用

こんな道具を使います

Jupyter Notebook

- そのままコードを実行できるUI
(ターミナルにコピー不要)
- マークダウンで文章も書ける
(Wiki同様の情報共有ツールとして機能)
- 実行結果が保存できる
(作業証跡として機能)

⇒実行できる手順書として機能する



NII

こんなやり方をします(今のところ)

構築・運用のために拡張された Jupyter Notebook

The screenshot shows a Jupyter Notebook interface with several annotations in Japanese:

- セルの実行** (Cell Execution): A callout pointing to the 'Run' button.
- 設計情報、運用状態などの文章** (Text for design information, operational status, etc.): A callout pointing to a text cell.
- 実行結果はセル単位でファイルに保存** (Execution results are saved per cell to the file): A callout pointing to the 'Save' button.
- 前提条件を満たしていないと失敗する確認用のセル** (Confirmation cell that fails if preconditions are not met): A callout pointing to a cell with a red error message.
- 実行結果の表示は要約版特定の文字列を含むか調べて表示** (Execution results are displayed by checking if they contain specific strings): A callout pointing to a cell with a red error message.
- コード化された作業内容** (Code-based work content): A callout pointing to a code cell.
- 以前の実行結果(お手本)** (Previous execution results (examples)): A callout pointing to a cell with a red error message.
- 実行結果をお手本と比較** (Compare execution results with examples): A callout pointing to a cell with a red error message.
- 重複実行を防ぐ凍結** (Freeze to prevent duplicate execution): A callout pointing to a cell with a red error message.
- 成功は緑失敗は赤でセルの処理状態が分かる** (Success is green, failure is red, so the cell's processing status is clear): A callout pointing to a cell with a red error message.
- 実行毎にファイルに保存(3回実行したら3つのファイル)** (Save to file for each execution (3 files after 3 executions)): A callout pointing to the 'Save' button.
- Notebook間の関係の図示** (Diagram showing relationships between notebooks): A callout pointing to a diagram.
- 作業テンプレートとなるNotebookを複製して、作業日時を接頭辞に付し、Notebookベースの作業を行う** (Replicate a notebook template, add work date/time as a prefix, and perform work based on the notebook): A callout pointing to a cell with a red error message.
- 作業後に残ったNotebookがそのまま証跡となる** (Notebooks remaining after work become evidence): A callout pointing to a cell with a red error message.
- 操作対象や再利用の単位でNotebookを分割** (Divide notebooks by operation target or reuse unit): A callout pointing to a cell with a red error message.

■ LC4RI:

Literate Computing for Reproducible Infrastructure

<https://www.janog.gr.jp/meeting/janog41/program/sp8lcri>

実際にやってみると：

Jupyter Notebook + SSH の辛いところ

現場の声：

端末へのコマンドを大きく書き換える必要があり障壁が高い

従来型手順書

事前準備：SSH端末より、user で host にログイン

```
% cd /etc/nginx/conf.d; ls -l  
% sudo tar cfz backup-`hostname`-`date +%F`.tgz *.conf
```



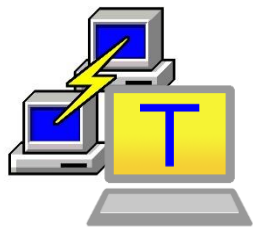
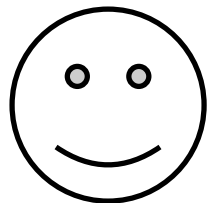
IPython Kernelの外部コマンド実行に変換

- a.** 外部コマンド実行記号(!)と、sshコマンドの挿入
- b.** 環境変数と\$CWDの復元
- c.** シェルのメタ文字の評価を変えないよう、エスケープ

Jupyter Notebook

```
a !ssh -l {user} {host} "cd /etc/nginx/conf.d; ls -l"  
!ssh -l {user} {host} b ¥  
"cd /etc/nginx/conf.d; ¥  
sudo tar cfz backup-¥`hostname¥`-¥`date +%F¥`.tgz *.conf"  
c
```

なぜこうなる



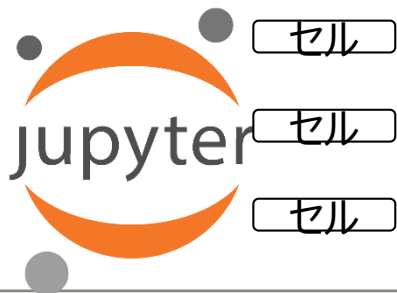
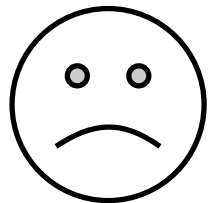
ずっと接続

SSH bash

コマンド

コマンド

コマンド



セル

セル

セル

Jupyter
カーネル

いちいちSSH

SSH

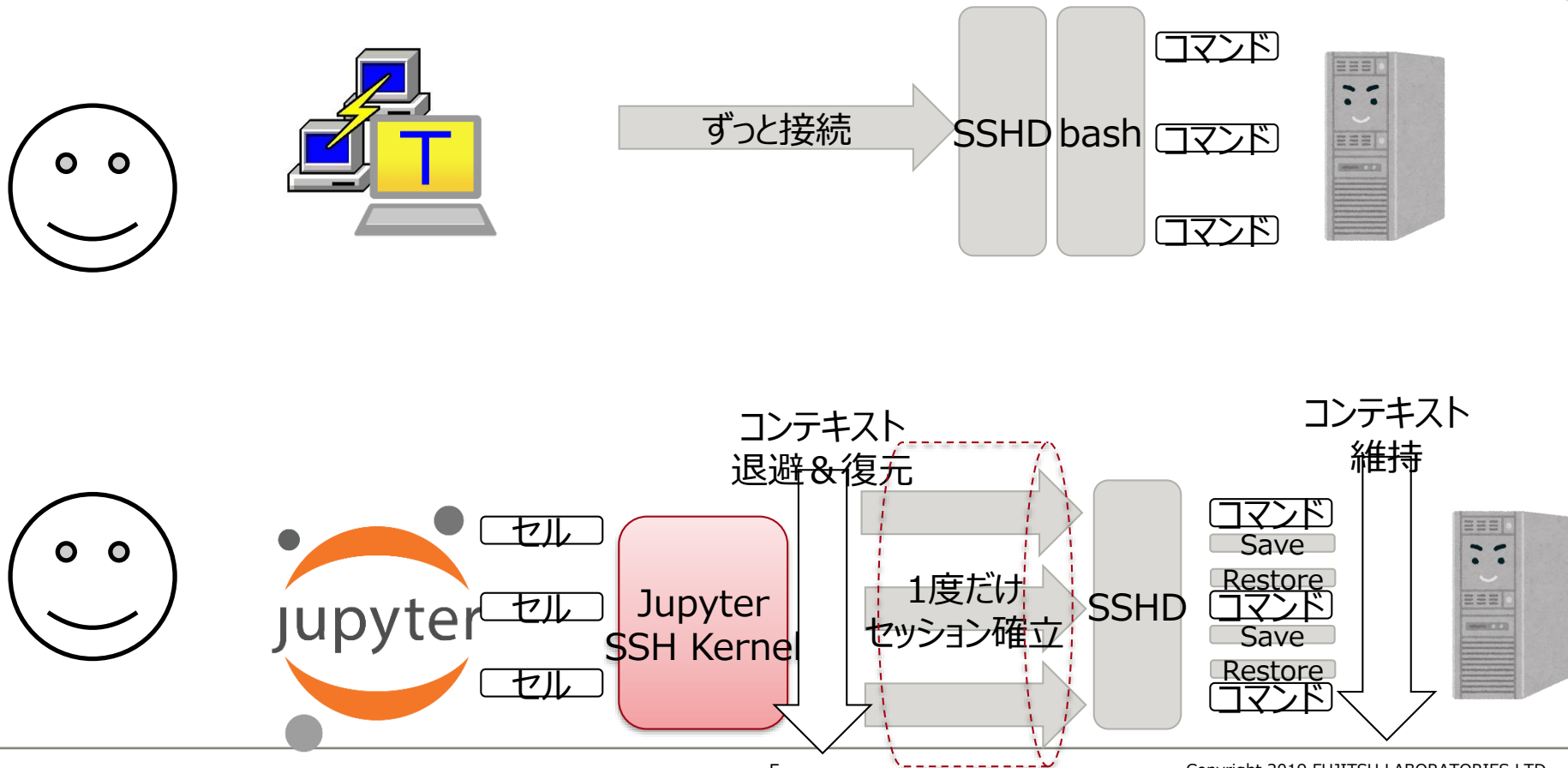
別コンテキスト

コマンド

コマンド



Jupyter + SSH Kernel



- LC4RI / Jupyterによる運用 をみんなにやさしく

- SSHにもやさしく

 - ~/.ssh/configコンパチ

 - SSHエージェントフォワード

- 導入もやさしく

 - `pip install -U sshkernel`

 - `python -m sshkernel install [--user|--sys-prefix]`

 - <https://github.com/NII-cloud-operation/sshkernel>