



イチから体験、 NGINX ハンズオントレーニング ～ 冗長構成編 ～

東京エレクトロン デバイス株式会社

2025年2月28日

F5製品に関する取り組み

- 1999年（メーカー日本法人ができる前からの一次代理店）
- F5国内販売額9年連続No.1の一次代理店
- 各ブランドに関するPoC、ハンズオントレーニング、構築支援サービス、日本語ヘルプデスクの提供
 - F5 BIG-IP
 - F5 NGINX
 - F5 Distributed Cloud Services
- 保守契約ユーザーに対する手厚い付加価値サービス
 - 会員制Webサポートサイトの提供(製品FAQ、各種ドキュメント等)
 - F5製品の重要情報をPush型でメール配信
(脆弱性、既知の重大不具合及び改修情報、リリース情報等)

営業支援

技術支援

F5
国内販売額
9年連続
No.1

幅広い
ラインナップ

F5
専任体制

● F5 資格取得者数

- 401 Security Solution Expert x 2
- 402 Cloud Solution Expert x 3
- 301b LTM Specialist x 8
- 302 DNS Specialist x 6
- 303 ASM Specialist x 3
- 304 APM Specialist x 4
- NGINX Management x 1
- NGINX Configure: Knowledge x 1



- ハンズオン環境の準備
- NGINX Plus について
- NGINX Plus の冗長構成
 - 高可用性 (High-Availability)
 - 設定同期
 - ステータス同期
- Key Value Store



環境の準備

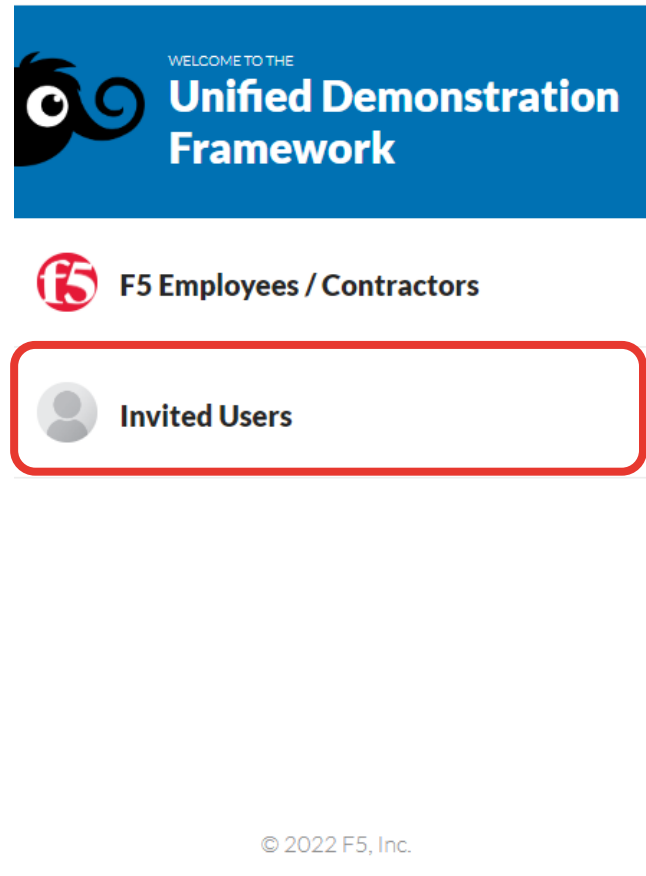
ハンズオン環境 (UDF環境)

<https://udf.f5.com>

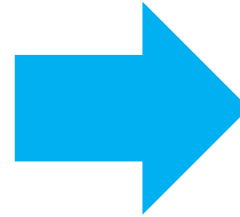
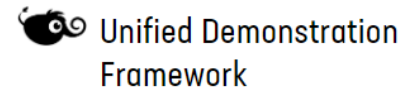
利用手順

https://20110926.fs1.hubspotusercontent-na1.net/hubfs/20110926/%E9%85%8D%E5%B8%83%E8%B3%87%E6%96%99/UDF%E5%88%A9%E7%94%A8%E6%89%8B%E9%A0%86_20240724%201.pdf

“Invited Users” を選択します



登録したユーザーでログインします



サインイン

ユーザー名

パスワード

サインイン

[パスワードをお忘れですか？](#)
[アカウントをお持ちでないですか？](#)
[サインアップ](#)

2段階認証 します

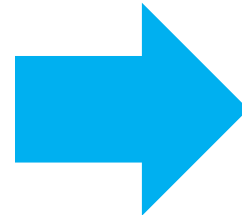


GOOGLE AUTHENTICATOR

Google Authenticatorのパスコードを入力します

コードを入力します

確認



ポリシーに Agree します

※ 事前に実施されている場合は表示されません

Welcome ttsuji

Privacy Policy

+

Terms and Conditions

+

Cookie Policy

+

☒ AGREE & CONTINUE

Privacy Policy, Terms and ConditionsおよびCookie Policyを確認し、
"AGREE & CONTINUE"にチェックを入れてクリック

トレーニング・セッションへの参加

招待されているトレーニング・セッションを選択して、
"LAUNCH"をクリック



 COURSE SESSIONS

Location
Any

Happening now

Date & Time	Course	Location	Instructors	
Fri 01 May 3:00 PM - 5:00 PM JST Duration: 2 hours	BIG-IP LTM hands-on in Tokyo	https://f5networks.zoom.us/j/6126548210	 Tetsuya TSUJI	UNREGISTER → LAUNCH



"Join"をクリック



LOBBY

BIG-IP LTM hands-on in Tokyo

FRI 1 MAY 03:00 PM TO FRI 1 MAY 05:00 PM JST

Ends in 2 hours <https://f5networks.zoom.us/j/6126548210>

[Join](#)

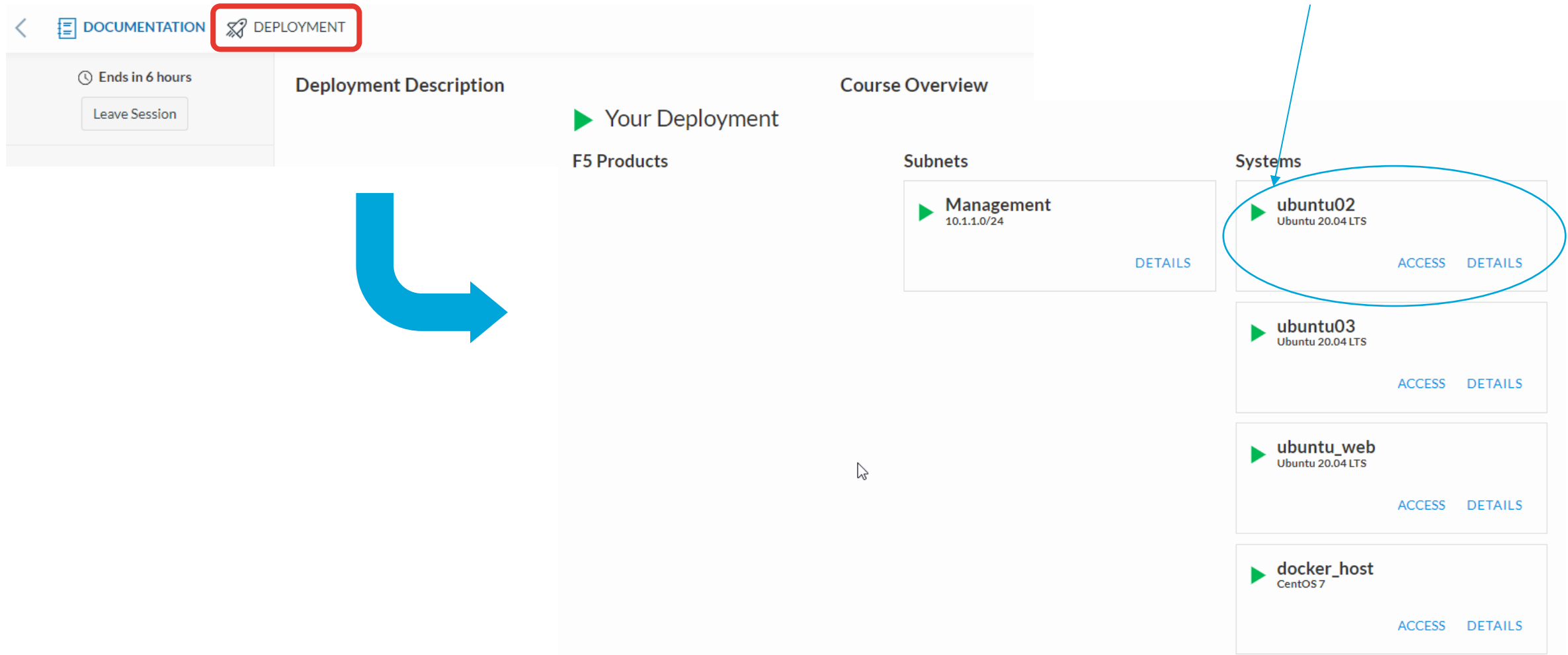
External Access Enabled

Course Overview
Hands-on course for customers/partners in Japan

Tools
[Manage SSH Keys](#)

Instructors
 Tetsuya TSUJI
F5, Solutions Engineer III

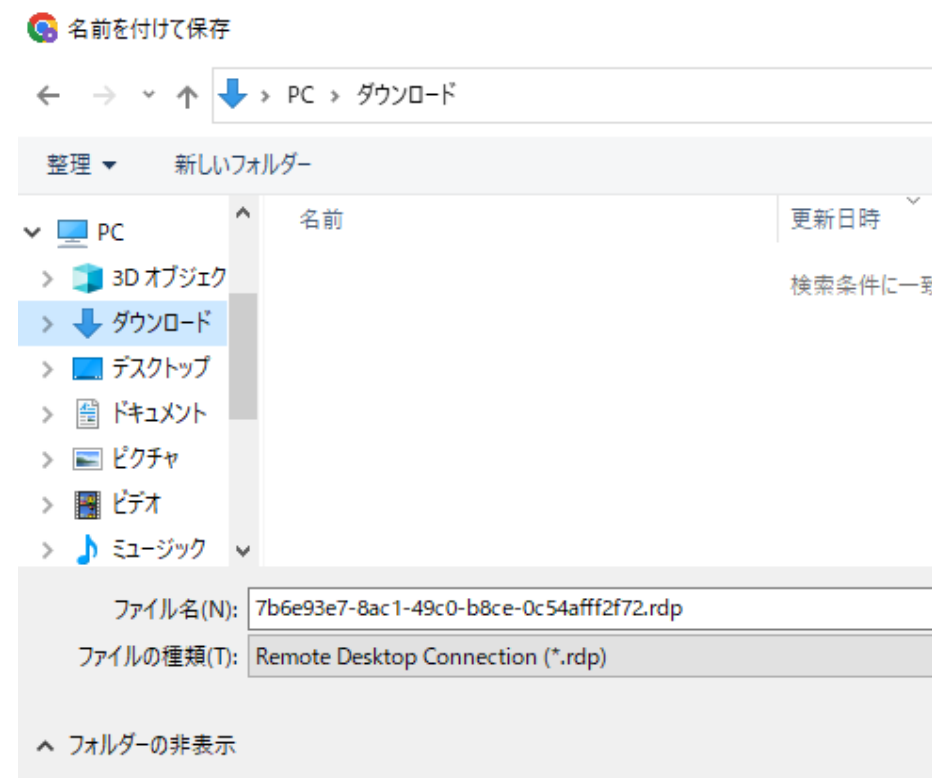
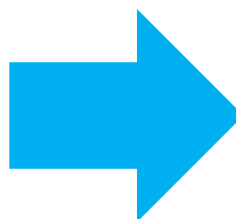
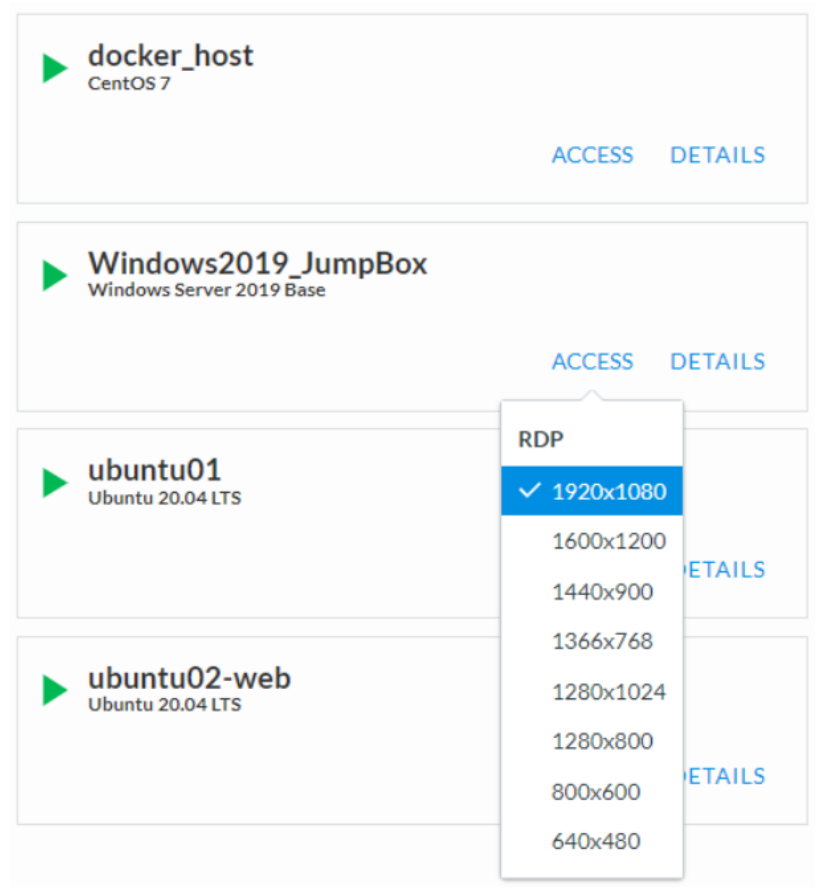
DEPLOYMENT タブを選択します



The screenshot shows a web interface for deployment management. At the top, there are two tabs: 'DOCUMENTATION' and 'DEPLOYMENT'. The 'DEPLOYMENT' tab is highlighted with a red rectangle. Below the tabs, there is a 'Deployment Description' section with a clock icon and the text 'Ends in 6 hours' and a 'Leave Session' button. To the right, there is a 'Course Overview' section. Below this, there are three main sections: 'Your Deployment', 'Subnets', and 'Systems'. The 'Your Deployment' section shows 'F5 Products'. The 'Subnets' section shows a 'Management' subnet with IP '10.1.1.0/24'. The 'Systems' section shows four systems: 'ubuntu02' (Ubuntu 20.04 LTS), 'ubuntu03' (Ubuntu 20.04 LTS), 'ubuntu_web' (Ubuntu 20.04 LTS), and 'docker_host' (CentOS 7). Each system has 'ACCESS' and 'DETAILS' links. A blue arrow points from the 'DEPLOYMENT' tab to the 'Systems' section. A blue circle highlights the 'ubuntu02' system, and a blue arrow points from the text 'すべての system が 緑△ になれば準備完了です' to this circle.

すべての system が 緑△ になれば準備完了です

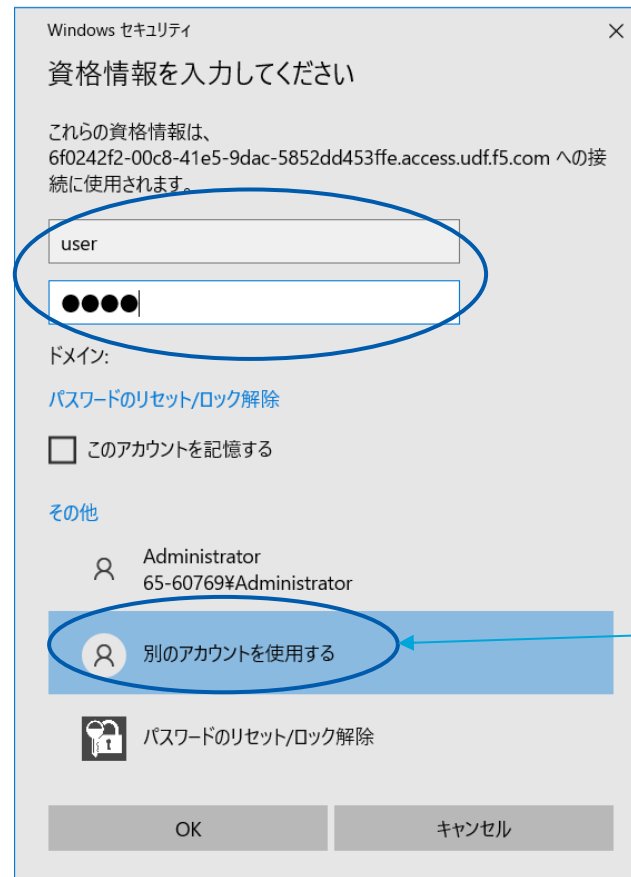
Windows2019_JumpBox の Access を選択して
RDP の画面サイズを選択して、RDP ファイルをダウンロードします



RDP ファイルを実行して windwos2019_jumpbox にログインします

User : user

Password : user

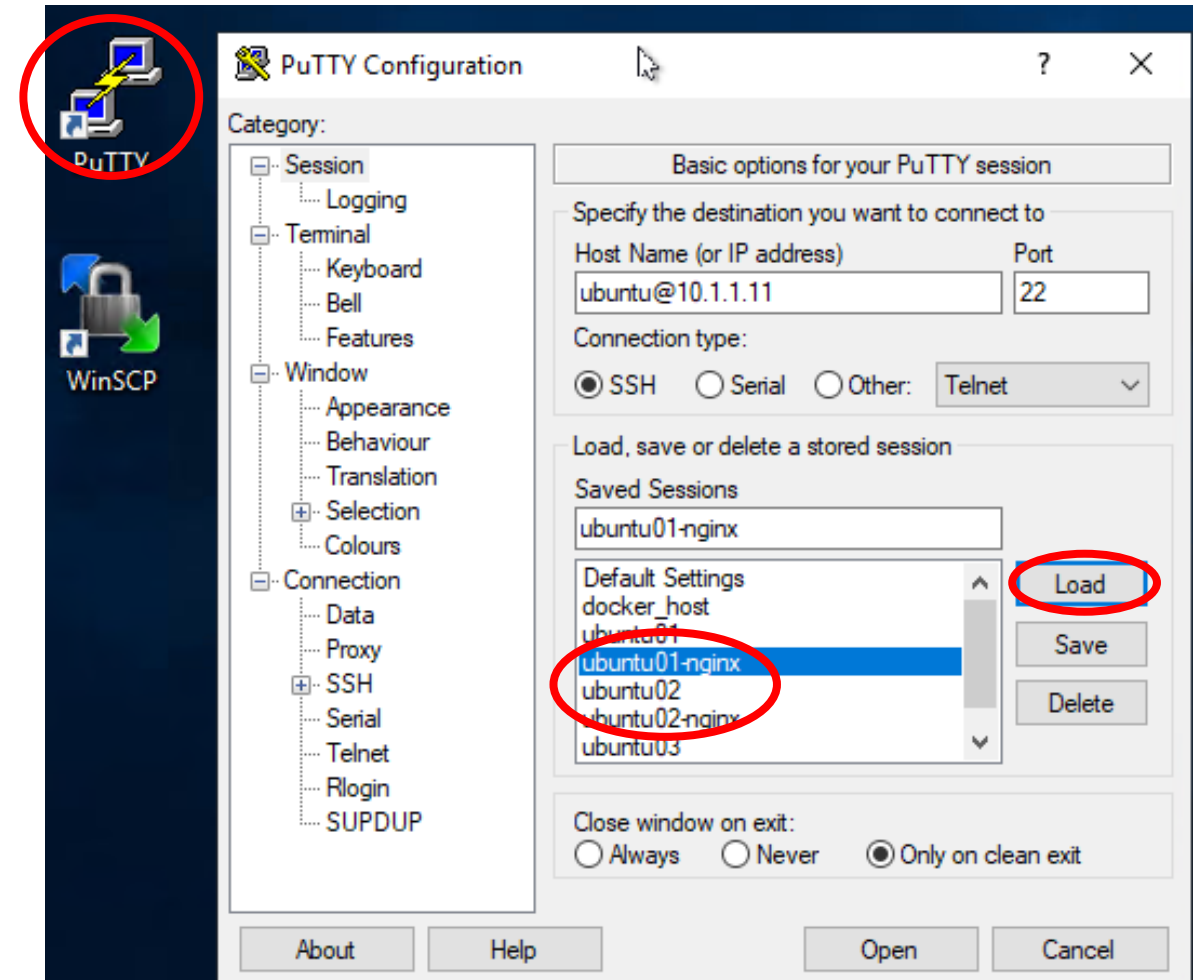


デフォルトは Administrator アカウントになっているため、こちらを選択します。

Putty を開いて **ubuntu01-nginx** と **ubuntu02-nginx** にログインします

- 1) Putty アイコンを選択します
- 2) ubuntu01-nginx を選択します
- 3) Load ボタンを選択します
- 4) Open ボタンを選択するとログインできます。
- 5) 同様に ubuntu02-nginx にログインします

※ ubuntu01/02 には NGINX がインストールされていないため
-nginx がついているホストを選択してください



事前準備 (ubuntu01-nginx / ubuntu02-nginx)

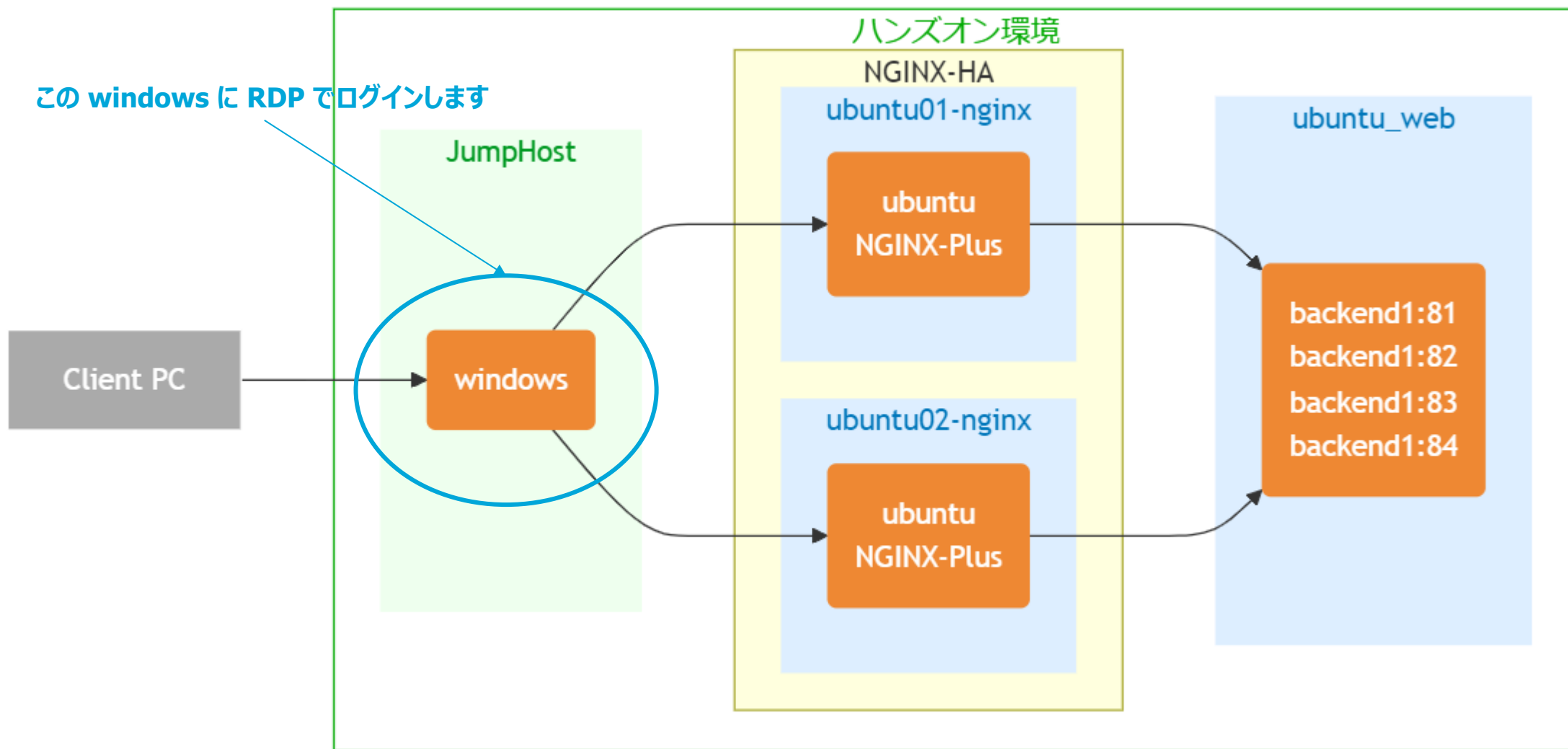
ubuntu01-nginx / ubuntu02-nginx の両方で以下を実施してください

- NGINX Plus がインストールされていることの確認

```
nginx -v
```

- ハンズオン設定集のダウンロード

```
cd ~/  
git clone https://github.com/araihub/f5j-nginx-plus-lab2-conf
```





簡単なおさらい

NGINX Plus の設定について (Tips)

upstream block (バックエンドサーバーをまとめたグループ)

server : 転送先サーバーの指定

zone : ワーカープロセス間の情報共有

server block (バーチャルホスト設定)

listen : 待ち受けIP:port

location : リクエストの待ち受け

status_zone : ダッシュボード用の共有メモリ

proxy_pass : 通信の転送先

※upstream 名を指定した場合、upstream のサーバーに転送

```
upstream server_group {  
    zone backend 64k;  
  
    server backend1:81;  
    server backend2:81;  
}
```

```
server {  
    listen 80;  
    status_zone server;  
    location / {  
        status_zone root;  
        proxy_pass http://server_group;  
    }  
}
```

設定ファイルで変数を使用

```
location / {  
    proxy_set_header X-Forwarded-For $remote_addr  
    proxy_pass http://backend;  
}
```

主な変数

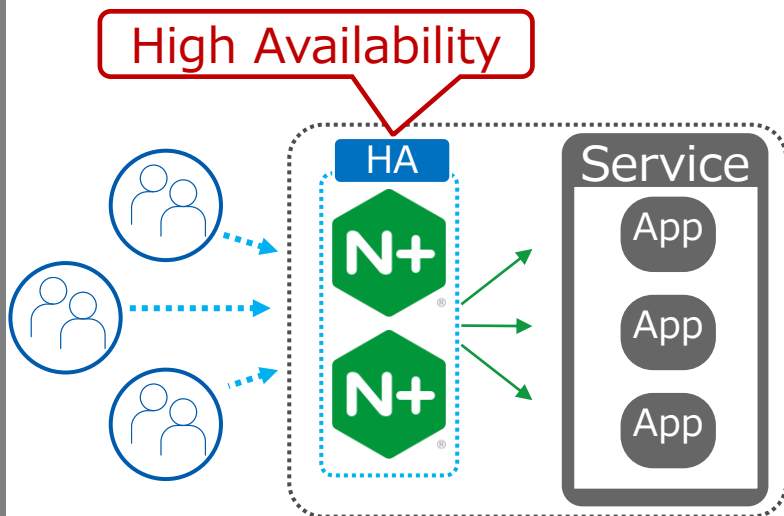
変数名	概要
\$remote_addr	クライアント(送信元)の IP アドレス
\$request_uri	リクエストの URI 情報 http://xxxxxxx/main の「/main」の部分
\$arg_<パラメータ>	リクエストの パラメータ情報 http://xxxxxxx/main?uid=123 の「123」の部分
\$http_<ヘッダ名>	HTTP リクエストのヘッダの値
\$cookie_<cookie名>	リクエストに含まれる <cookie名> の値



NGINX の冗長構成

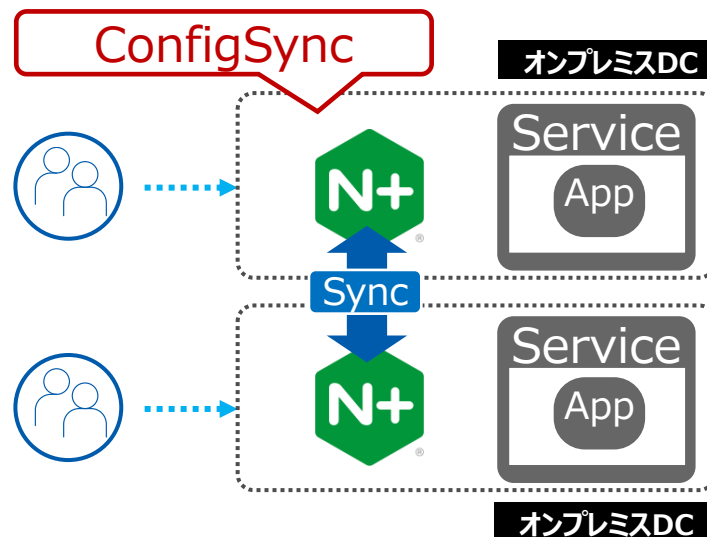
高可用性 (HA)

nginx-ha-keepalived



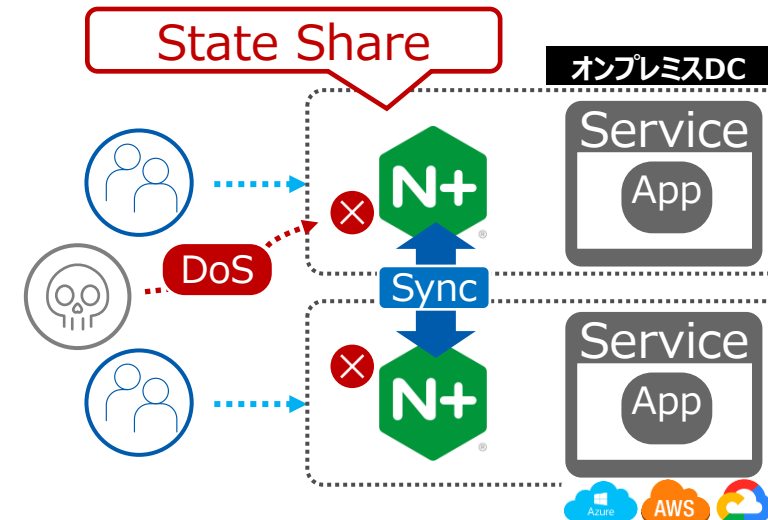
設定同期 (configsync)

nginx-sync



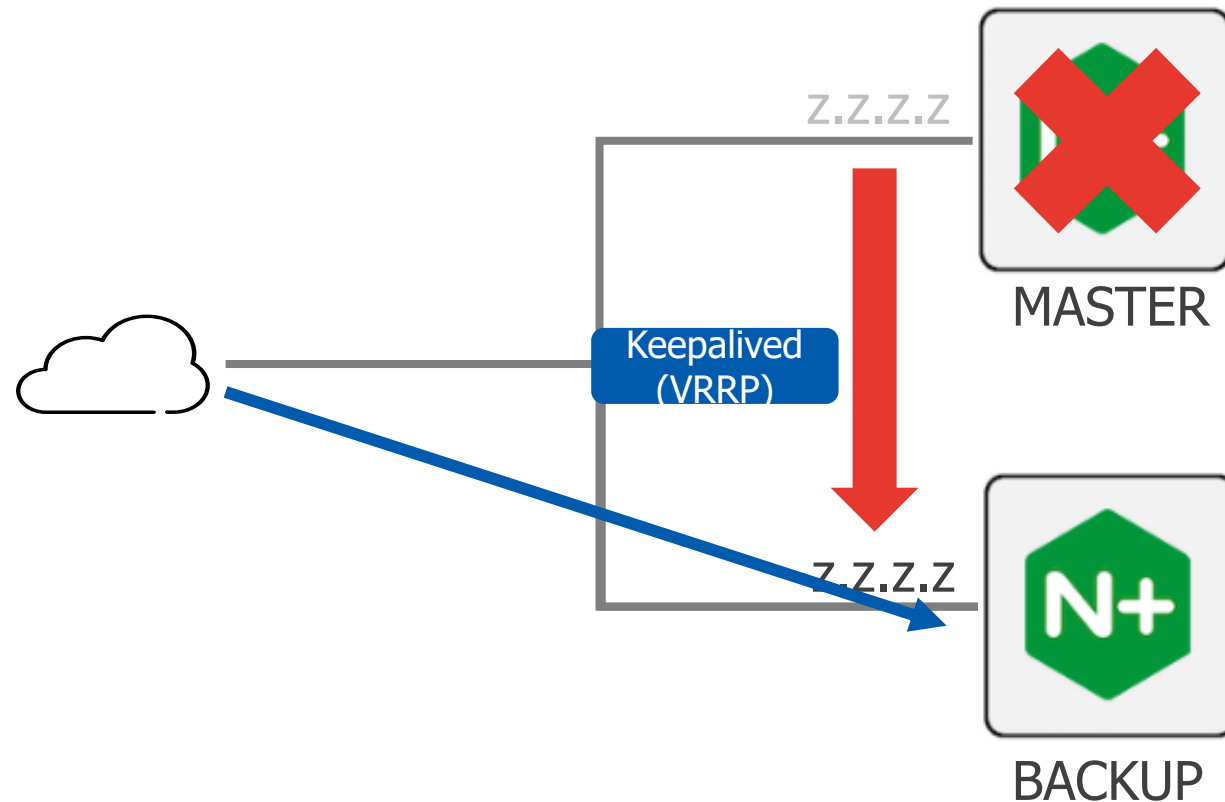
ステータス同期 (zonesync)

zone_sync



高可用性 (High Availability)

- nginx-ha-keepalived パッケージ
 - NGINX Plus 専用にカスタマイズされた keepalived
 - 対話形式で keepalived の config を生成
- Keepalived とは
 - VRRP による高可用性
 - IP アドレスの動的切り替え
 - サービスの状態監視



対話形式でkeepalivedのConfig作成

```
sudo nginx-ha-setup
```

Step 1: configuring internal management IP addresses.

Do you want to use this address for internal cluster communication? (y/n)

IP address of this host is set to: **10.1.1.7/24**

Primary network interface: **ens5**

Now please enter IP address of a second node: **10.1.1.6**

You entered: **10.1.1.6**

Is it correct? (y/n)

IP address of the second node is set to: **10.1.1.6**

Step 2: creating keepalived configuration

Enter cluster IP address: **10.1.1.100**

You entered: 10.1.1.100

Is it correct? (y/n)

Please choose what the current node role is:

1) MASTER

2) BACKUP



/etc/keepalived/keepalived.conf

```
vrrp_instance VI_1 {  
    interface                ens5  
    priority                  101  
    virtual_router_id        51  
    advert_int                1  
    accept  
    garp_master_refresh       5  
    garp_master_refresh_repeat 1  
    unicast_src_ip            10.1.1.7/24  
  
    unicast_peer {  
        10.1.1.6  
    }  
  
    virtual_ipaddress {  
        10.1.1.100  
    }  
}
```

/etc/keepalived/keepalived.conf

```
vrrp_instance VI_1 {  
    interface ens5  
    priority 101  
    unicast_src_ip 10.1.1.7/24  
    unicast_peer {  
        10.1.1.6  
    }  
    virtual_ipaddress {  
        10.1.1.100  
    }  
    track_script {  
        chk_nginx_service  
        chk_manual_failover  
    }  
    notify "/usr/lib/keepalived/nginx-ha-notify"  
}
```

.. 一番大きな値を持っているものが MASTER になる

.. MASTER が持つ Virtual IP

.. Priority を変動させる要素

/etc/keepalived/keepalived.conf

```
vrp_script chk_manual_failover {  
    script "/usr/lib/keepalived/nginx-ha-manual-failover"  
    interval 10  
    weight 50  
}
```

.. 手動切り替え用スクリプト

```
vrp_script chk_nginx_service {  
    script "/usr/lib/keepalived/nginx-ha-check"  
    interval 3  
    weight 50  
}
```

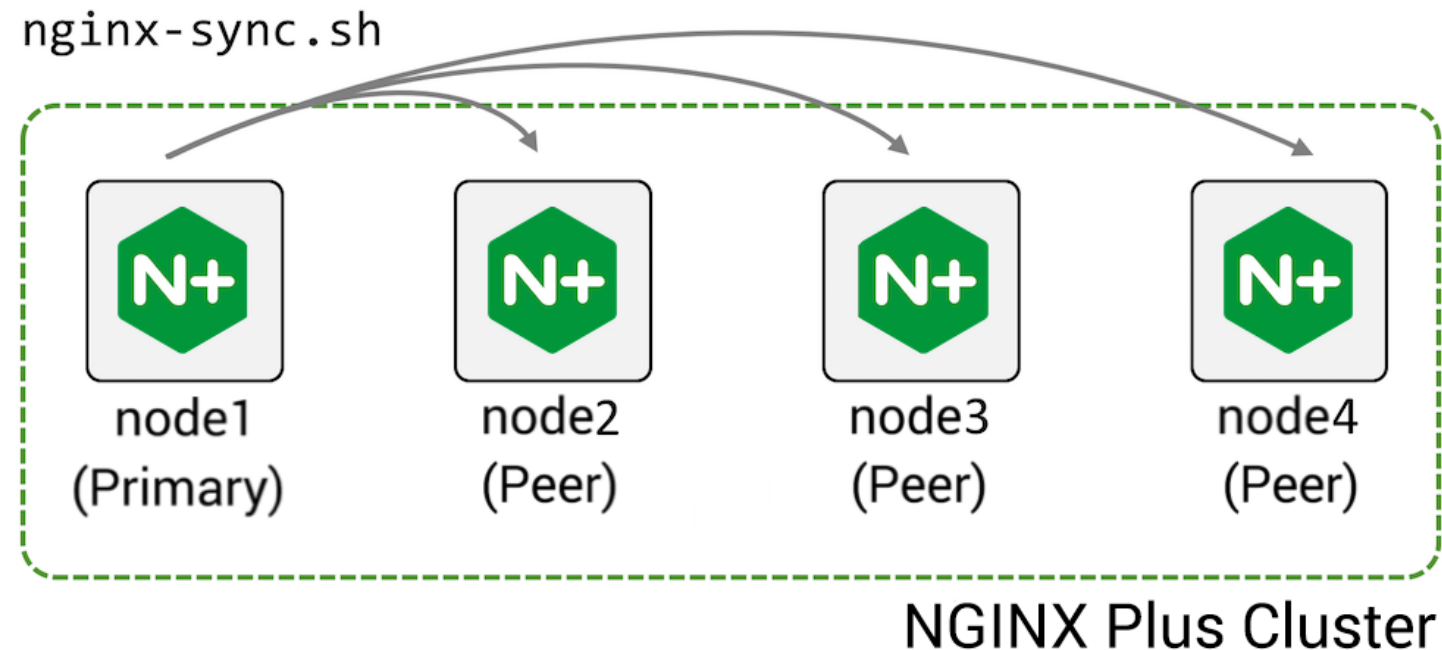
.. NGINX Process の死活監視スクリプト

```
vrp_instance VI_1 {  
<中略>  
}
```



NGINX Plus の設定同期

- nginx-sync パッケージ
 - NGINX Plus 専用の設定同期スクリプト
 - rsync を利用して、Primary の設定を Peer に同期
 - 同期先と同期ファイルを複数指定可能



- Primary – Peer 間で root ユーザーの SSH 鍵認証
 - Primary で実施

```
# sudo ssh-keygen -t rsa -b 2048  
  
# sudo cat /root/.ssh/id_rsa.pub ssh-rsa  
AAAAB3Nz4rFgt...vgaD root@centos01
```

- Peer で実施

```
# sudo mkdir /root/.ssh  
  
# sudo echo `from="centos01" ssh-rsa AAAAB3Nz4rFgt...vgaD  
root@nginx01` >> /root/.ssh/authorized_keys
```

/etc/nginx-sync.conf

```
NODES="10.1.1.6"  
CONFPATHS="/etc/nginx/ssl /etc/nginx/conf.d"  
EXCLUDE="dummy.conf"
```

- NODES : 設定を同期させるノード (Peer ノードを指定)
- CONFPATHS : 同期させるディレクトリ/ファイルのpath
- EXCLUDE : 同期させない(除外する)ファイル

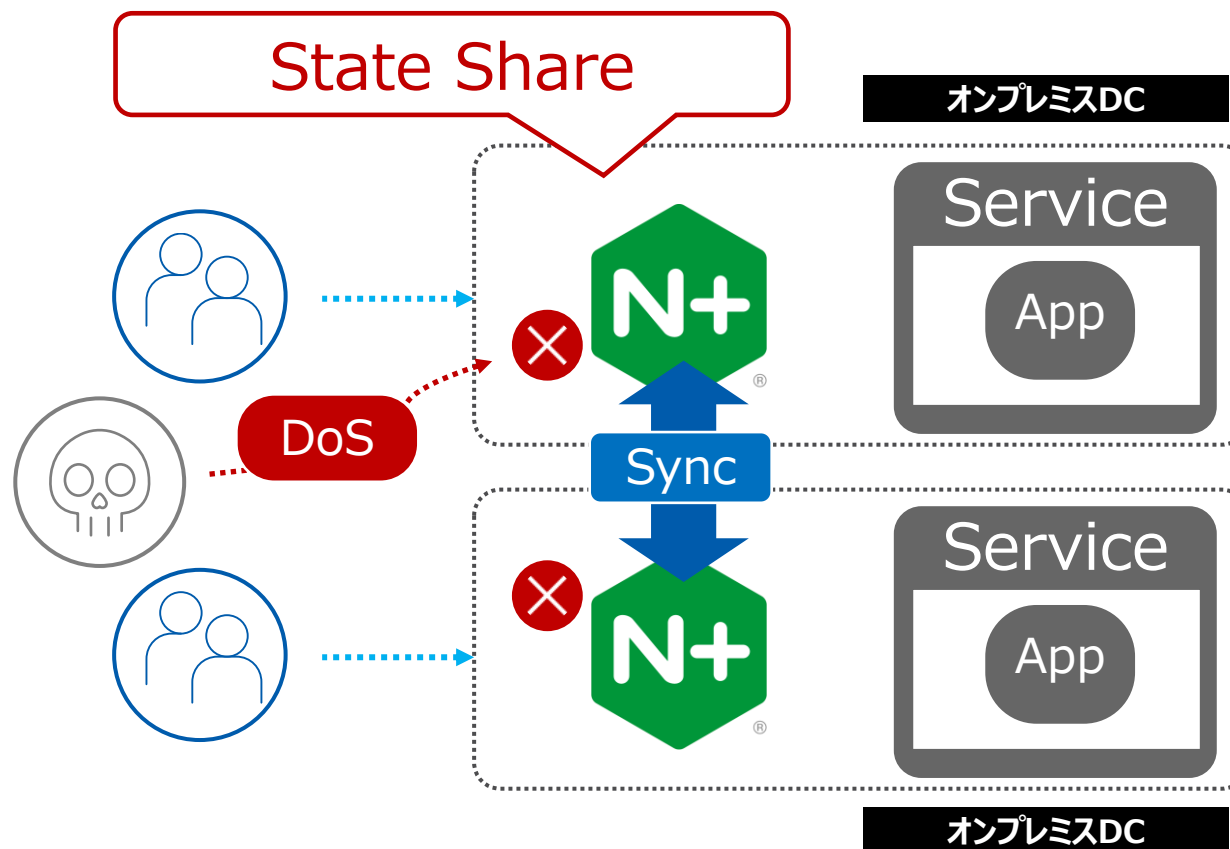
※ 複数指定する場合は、スペースで区切ります



NGINX Plusのステータス同期

ステータス同期 (zone sync)

- 複数のNGINXでリクエストを処理する際の整合性を確保
- リアルタイムに変化する以下の情報(共有メモリゾーン) を同期
 - レート制限
 - Sticky Lean 情報
 - Key Value Store



/etc/nginx/conf.d/stream.conf (zone sync 用の設定)

```
stream {  
    server  
        listen 9000;  
  
        zone_sync;                .. zone sync の有効化  
        zone_sync_server 10.1.1.6:9000;    .. Zone の同期先を指定  
    }  
}
```

/etc/nginx/conf.d/default.conf

```
limit_req_zone $remote_addr zone=req:1M rate=1r/m sync;
```

```
sticky learn zone=sessions:1m create=$upstream_cookie_session lookup=$cookie_session sync;
```



Key Value store

Key Value Store

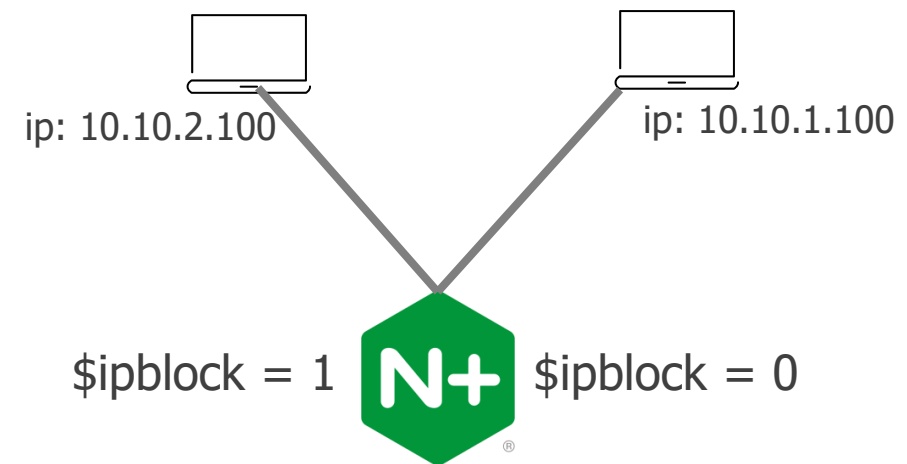
動的に処理可能 な Key-Value 型のデータベース

共有メモリ上で管理され API を使用して CRUD 操作が可能

Value 変数は Key とマッチした値を返す

Key (\$remote_addr)	Value (\$ipblock)
10.10.1.100	0
10.10.2.100	1

\$ip_block の値は Key (クライアントのIPアドレス) は
10.10.1.100 の場合 0
10.10.2.100 の場合 1



keyval / keyval_zone

書式 : keyval_zone zone=<name>:<size> ;

書式 : keyval <Key> <\$variable> zone=<name> [timeout=*time*] [sync];

- Zone : key-Value のペアを管理する共有メモリゾーン
- <key> : 条件を指定する入力データの指定 (ホスト名やIPアドレスなど)
- \$variable : 任意の変数。条件に対応する出力や動作 (サーバ、制御フラグなど)
- Timeout : Key-Value のペアが削除されるまでの時間 (オプション)
- Sync : ステータス同期する場合に使用 (オプション)

- レコードの追加

```
curl -s 127.0.0.1:8080/api/8/http/keyvals/ipblock -X POST -d '{ "127.0.0.1": "1" }'
```

- レコードを参照

```
curl -s 127.0.0.1:8080/api/8/http/keyvals/ | jq .
```

```
{
  "ipblock": {
    "127.0.0.1": "1"
  }
}
```

- レコードを更新

```
curl -s 127.0.0.1:8080/api/8/http/keyvals/ipblock -X PATCH -d '{ "127.0.0.1": "0" }'
```

/etc/nginx/conf.d/default.conf

```
keyval_zone zone=ipblock:1M ;
keyval $remote_addr $block zone=ipblock ;

server {
    location / {
        if ($block) {
            return 403;
        }
        proxy_pass http://backend_app;
    }
}
```

ipblock zone メモリ上に保存されているリスト

```
curl -s 127.0.0.1:8000/api/9/http/keyvals/ipblock
{"ipblock": {
    "10.0.0.2" : 1,
    "10.0.0.3" : 1,
}}
```

/etc/nginx/conf.d/default.conf

```
keyval_zone zone=iplist:32k ;  
keyval $arg_user $permit_ip zone=iplist;
```

```
server {  
    listen 80;  
    location / {  
        if ( $permit_ip = "" ) {  
            set $permit_ip $remote_addr;  
        }  
  
        if ( $remote_addr != $permit_ip ) {  
            return 403 "Mismatch client IP address";  
        }  
  
        proxy_pass http://server_group;  
    }  
}
```



まとめ

以下についてご紹介しました。

- NGINX Plus について
- NGINX Plus の冗長構成
 - 高可用性 (High-Availability)
 - 設定同期
 - ステータス同期
- Key Value Store



参考情報

NGINXがまるっと分かる！ブログ更新中！

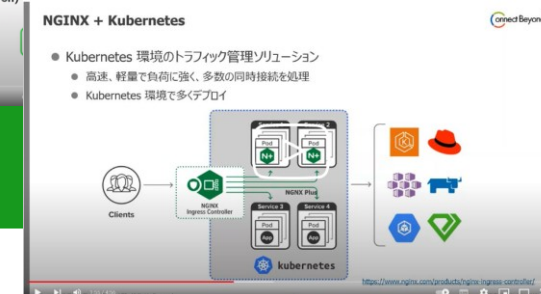
<https://cn.teldevice.co.jp/blog/search/?q=NGINX>

ブログ

YouTube

NGINXを簡単解説！

F5 NGINX - YouTube



無償トライアルと個別勉強会のご案内

NGINX Plus 無償トライアルライセンス



個別勉強会

イチから体験！
NGINX ハンズオントレーニング
～ 基礎編 ～



NGINX
Part of F5



<https://cn.teldevice.co.jp/product/f5-nginx/>

**今後のより充実したセミナー開催のため
アンケートへのご協力をお願いいたします。**

**配布資料は、セミナー終了後にお送りするメールにて
ご案内いたします。**

**長時間にわたりご視聴いただき
誠にありがとうございました。**