

OpenSource/Linux技術文書



HP SmartStart Scripting Toolkit for Linux 活用術

～ ProLiantサーバーのBIOS/RAID/iLOチップ設定情報の吸い出しと一斉配布 ～

日本ヒューレット・パッカード株式会社
ESSNプリセールス統括本部
エンタープライズ・サーバー・ストレージ技術第一本部
Linuxソリューション部 古賀 政純

2012年2月23日

目次

[本ドキュメントについて]	4
PXE ブートと HP SSSTK を組み合わせたハードウェア設定/配布サーバーについて	5
DHCP サーバーの設定	6
TFTP サーバーの設定	6
NFS サーバーの設定	7
SSSTK の設定	8
BIOS 設定情報を吸い出す SSSTK のスクリプト作成	10
BIOS 設定情報を配布する SSSTK のスクリプト作成	10
[注意] XML ファイルが必要なサーバー機種対応の conrep と SSSTK スクリプト	10
[注意] 管理対象サーバー機種が XML ファイルを必要とする場合の conrep の入手と配備	11
HP SmartArray 設定情報を吸い出す SSSTK のスクリプト作成	13
HP SmartArray 設定情報を配布する SSSTK のスクリプト作成	13
HP iL03 設定情報を吸い出す SSSTK のスクリプト作成	13
HP iL03 設定情報を配布する SSSTK のスクリプト作成	14
[注意] HP L0100 の設定情報を配布する SSSTK のスクリプト作成	14
BIOS/SmartArray/iL03 設定情報の吸い出しをバッチ処理するスクリプト作成	14
BIOS/SmartArray/iL03 設定情報の配布をバッチ処理するスクリプト作成	15
BIOS、HP SmartArray RAID コントローラー、iL03 の設定情報の吸い出し	16
BIOS、HP SmartArray RAID コントローラー、iL03 の設定情報の配布	17

図表目次

図 1. HP SSSTK サーバーによる HW 情報の吸い出しと配布.....	5
図 2. HP ProLiant DL2000 は 2U で 4 ノードが稼働する高密度サーバー.....	11
図 3. HP ProLiant SL6500 は 4U で 8 ノードが稼働する高密度サーバー.....	11
図 4. HP ProLiant DL180G6 は 2U で 内蔵ディスク 42TB の容量を持つ.....	11
図 5. 管理対象サーバーの PXE ブートの様子.....	16
図 6. 管理対象サーバー上で SSSTK がオンメモリで実行されている様子.....	16
図 7. 管理対象サーバー上でオンメモリで実行されている SSSTK が NFS サーバーの /nfsroot/data_files ディレクトリに bios.cfg ファイルをアップロードしていることがわかる.....	17
図 8. HP ProLiant DL360G7 の POST 画面で Hyper-threading が disabled になっている旨の表示が見える。PXE/NFS/SSSTK サーバーによってリモートから BIOS を無人で設定できたことがわかる.....	19
図 9. HP ProLiant DL360G7 の BIOS 画面で CPU の仮想化機能が「無効」になっている旨の表示が日本語表示されている。PXE/NFS/SSSTK サーバーによって遠隔から BIOS を無人で設定できたことがわかる.....	19

[本ドキュメントについて]

- コマンドラインでの入力や表示が長く紙面の都合で折り返して記載する場合は、下記のように「¥」記号を挿入して複数行にわたって記載しています。複数行にわたって記載されていても実際には1行で入力、表示するものは、その記述の最後に「(実際には1行で入力)」や「(実際には一行で表示)」を挿入しています。

例) # ps auxw |grep dhcp |grep -v grep

```
dhcpcd 41913  0.0  0.4 31172 26420  ??  ls   7:15AM  0:00.01 /usr/local/sbin/dhcpcd -cf ¥  
/usr/local/etc/dhcpd.conf -lf /var/db/dhcpd (実際には一行で表示)
```

- 本ドキュメントの内容については充分チェックをしておりますが、その正確性を保証する物ではありません。また、将来、予告なしに変更することがあります。
- 本ドキュメントの使用で生じるいかなる結果も利用者の責任となります。日本ヒューレット・パッカード株式会社は、本ドキュメントの内容に一切の責任を負いません。
- 本ドキュメントの技術情報は、ハードウェア構成、OS、アプリケーションなど使用環境により大幅に数値が変化する場合がありますので、十分なテストを個別に実施されることを強くお勧め致します。
- 本ドキュメント内で表示・記載されている会社名・サービス名・商品名等は各社の商標又は登録商標です。
- 本ドキュメントで提供する資料は、日本の著作権法、条約及び他国の著作権法にいう著作権により保護されています。

本ドキュメントは、ネットワークカードのPXEブート機能とHP SmartStart Scripting Toolkit (以下HP SSSTK) を組み合わせて、HP ProLiantサーバーのハードウェア情報 (BIOS、RAIDコントローラー、遠隔管理チップ iLO3) の吸い出しと配布を行うためのサーバーの構築手順書です。PXEブートとHP SSSTKによるHP ProLiantサーバーのハードウェア情報の吸い出し/配布のサーバーを構築することにより、大量のHP ProLiantサーバーのハードウェア配備の手間を大幅に削減することができます。RHELの標準機能であるDHCP/TFTP/NFSの各種サービスの機能により、人間がキーボードやマウス操作によって行っていたHP ProLiantサーバーのBIOS/RAIDコントローラー/遠隔管理チップiLO3の設定を自動化することができます。

PXE ブートと HP SSSTK を組み合わせたハードウェア設定/配布サーバーについて

最近のHP ProLiantサーバーではPXEブートによるネットワーク経由でLinuxやFreeBSD等のインストールが可能となっていますが、この機能を応用しHP SSSTKをPXEブート経由で起動させることにより、HP ProLiantのBIOSやRAIDコントローラー等のハードウェア設定の自動化が可能です。HP SSSTKが提供するPXEブート用のLinuxイメージや各種ツールを読み込むことにより、BIOSやRAIDコントローラーの設定時に人間が介在するキーボードやマウス操作等を一切行わない自動化を実現することが可能です。大量のサーバーのBIOS設定やRAIDコントローラー設定を一斉に全自動で行いたい場合に有用です。

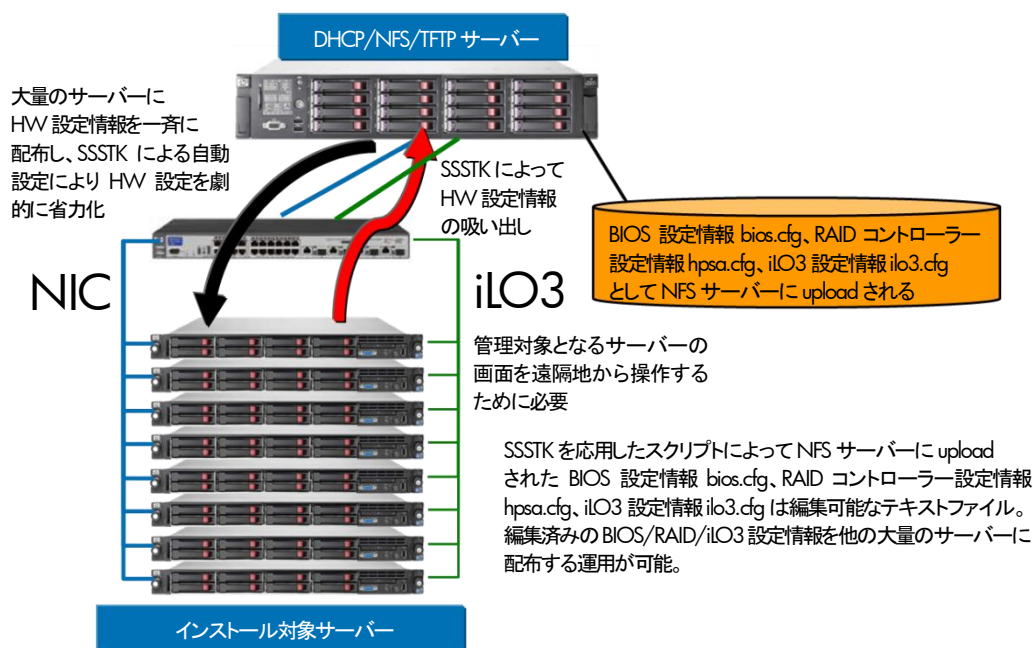


図 1. HP SSSTK サーバーによる HW 情報の吸い出しと配布

以下、PXEブートとHP SSSTKによるハードウェア情報の吸い出しと配布を実現するサーバーの作成手順です。本ドキュメントのSSSTKサーバーのOSはRHEL6.xです。SSSTKサーバーとなるHP ProLiantを用意し、RHEL6.xをあらかじめインストールしておいてください。また、DHCPサービス、NFSサービス、TFTPサービスが必要となりますので、それらサービスを提供するパッケージをRHEL6.x環境にインストールしておいてください。特にTFTPについてはtftp-server RPMパッケージのインストールが必要ですので注意して下さい。

DHCP サーバーの設定

DHCPサーバーの設定を行います。DHCPサーバーのIPアドレスを172.16.1.85/16とします。

```
# service dnsmasq stop
# chkconfig dnsmasq off
# vi /etc/dhcp/dhcpd.conf
ddns-update-style interim;
ignore client-updates;
subnet 172.16.0.0 netmask 255.255.0.0 {
    option routers                172.16.1.85;
    option subnet-mask            255.255.0.0;
    option domain-name            "jpn.linux.hp.com";
    option domain-name-servers    16.110.135.51;
    option time-offset            -18000;
    range dynamic-bootp          172.16.0.1 172.16.0.254;
    default-lease-time           21600;
    max-lease-time               43200;
    next-server                   172.16.1.85;
    filename                      "pxelinux.0";
}

# chkconfig dhcpd on
# service dhcpd start
dhcpd を起動中: [ OK ]
#
# ps auxw |grep dhcp |grep -v grep
dhcpd  28419  0.0  0.0  46508  3940 ?        Ss   18:00   0:00 /usr/sbin/dhcpd -user dhcpd
-group dhcpd
```

TFTP サーバーの設定

/etc/xinetd.d/tftpファイルのtftp行の-sオプション以降にTFTPサーバーが、遠隔にある管理対象サーバーに提供するディレクトリを指定しますが、RHELに標準で用意されている設定をそのまま利用することになります。ここではtftpファイルを確認します。

```
# cd /etc/xinetd.d/
# cat tftp
...
service tftp
{
    disable = no
```

```

        socket_type          = dgram
        protocol             = udp
        wait                 = yes
        user                 = root
        server               = /usr/sbin/in.tftpd
        server_args          = -s /var/lib/tftpboot
        per_source           = 11
        cps                  = 100 2
        flags                = IPv4
    }

```

上記より、-sオプションの後の/var/lib/tftpbootディレクトリ以下にPXEブート環境の各種設定ファイルを配置することになります。dhcpd.confファイルでfilenameに「pxelinux.0」を指定し、tftpファイルでserver_argsに「-s /var/lib/tftpboot」を指定しているため、TFTPサーバーは/var/lib/tftpboot/ディレクトリにpxelinux.0ファイルを配置する必要があります。pxelinux.0ファイルの入手と配備は後述します。まず、TFTPサービスを起動させます。

```

# chkconfig tftp on
# chkconfig xinetd on
# service xinetd reload
設定を再読み込み: [ OK ]

```

NFS サーバーの設定

NFSサーバー上にSSSTKを配備するディレクトリ/nfsrootを作成します。

```
# mkdir -p /nfsroot
```

NFSサーバーの設定を行います。NFSサーバーの/etc/exportsに設定します。/nfsrootディレクトリは管理対象となるProLiantサーバーのSSSTKの実行に必要とされます。

```
# vi /etc/exports
/nfsroot *(rw,no_root_squash)
```

NFSの関連サービスを起動します。

```

# service nfs start
NFS サービスを起動中: [ OK ]
NFS クォータを起動中: [ OK ]
NFS デーモンを起動中: [ OK ]
NFS mountd を起動中: [ OK ]
# chkconfig nfs on

```

NFSサービスによって/nfsrootディレクトリが提供されているか確認します。

```

# showmount -e localhost
Export list for localhost:
/nfsroot *

```

SSSTK の設定

SSSTKをNFSサーバー上に配備します。まずSSSTKを入手します。SSSTKはHPのダウンロードサイトから入手できます。

SmartStart Scripting Toolkit Linux Edition (SSSTK) の入手先：

https://support.hpe.com/hpsc/swd/public/detail?swItemId=MTX_7dd0bdbe0fa84ebdaeca21dc85

上記ダウンロードサイトからss-scripting-toolkit-linux-8.70.tar.gzを入手します。入手したss-scripting-toolkit-linux-8.70.tar.gzをNFSサーバーで展開します。

```
# pwd
/root
# tar xzvf ss-scripting-toolkit-linux-8.70.tar.gz
# cd ss-scripting-toolkit-linux-8.70/
# ls -F
LinuxUsersGuide8.70.pdf  conrep_xml/  extras/      mkusbkey.sh*  windows_unattend/
boot_files/              contrib/     linux_unattend/  scripts/
changelog                data_files/  mkiso.sh*      utilities/
```

上記全てのファイルとディレクトリをNFSサーバーの/nfsrootディレクトリにコピーします。data_filesディレクトリは空ディレクトリですが、必要となりますので、全てコピーします。

```
# cp -a * /nfsroot/
```

管理対象となるHP ProLiantサーバーがPXEブートするのに必要なファイルvmlinuz、initrd.img、pxelinux.0を置きます。

```
# mkdir /var/lib/tftpboot/ssstk870
# cd /nfsroot/boot_files/
# cp -a vmlinuz /var/lib/tftpboot/ssstk870/
# cp -a initrd.img /var/lib/tftpboot/ssstk870/
# cp -a pxelinux.0 /var/lib/tftpboot/
```

PXEブートに必要な設定ファイルdefaultを作成します。SSSTKが保存されているNFSサーバーのIPアドレスは172.16.1.85としますので、defaultファイル内では「sstk_mount=172.16.1.85:/nfsroot」と記述します。NFSサーバーの/var/lib/tftpboot/ssstk870ディレクトリに配置したvmlinuzファイルとinitrd.imgファイルを参照できるようにkernel行とappend行に相対パスで記述します。

```
# mkdir -p /var/lib/tftpboot/pxelinux.cfg
# cd /var/lib/tftpboot/pxelinux.cfg/
# vi default
default sstk870cap
##default sstk870dep
### SSSTK 8.70 Capture
label sstk870cap
kernel sstk870/vmlinuz
append initrd=ssstk870/initrd.img root=/dev/ram0 rw ramdisk_size=190272 ide=nodma ¥
ide=noraid pnpbios=off network=1 sstk_mount=172.16.1.85:/nfsroot sstk_mount_type=nfs ¥
```



```
sstk_mount_options=rw,nolock sstk_script=/capall.sh
### SSSTK 8.70 Deploy
label ssttk870dep
kernel ssttk870/vmlinuz
append initrd=ssttk870/initrd.img root=/dev/ram0 rw ramdisk_size=190272 ide=nodma ¥
ide=noraid pnpbios=off network=1 sstk_mount=172.16.1.85:/nfsroot sstk_mount_type=nfs ¥
sstk_mount_options=rw,nolock sstk_script=/depall.sh
```

RAIDコントローラーの設定の吸い出しと配布を行うhpacuscriptingツールを編集します。hpacuscriptingツールのHPACUCLI_BIN_INSTALLATION_DIR=行を下記のように設定します。

```
# cd /nfsroot/utilities/hpacucli/
# ls -lF
合計 2192
-r-x-----. 1 root root    556 12月  2 11:43 2010 hpacucli*
-rw-----. 1 root root   7602 12月  2 11:43 2010 hpacucli-8.70-8.0.noarch.txt
-rw-r--r--. 1 root root  20943 12月  2 11:43 2010 hpacucli.license
-r-x-----. 1 root root    586  1月 16 23:08 2012 hpacuscripting*
-rwxr-xr-x. 1 root root 2193229 12月  2 11:43 2010 libcpqimgr.so*
-r-x-x-x-x. 1 root root    899  3月 10 12:39 2011 mklocks.sh*

# cp hpacuscripting hpacuscripting.org
# vi hpacuscripting
...
HPACUCLI_BIN_INSTALLATION_DIR=. /
...
```

BIOS 設定情報を吸い出す SSSTK のスクリプト作成

BIOSの設定情報を吸い出すスクリプトcapbios.shは/var/lib/tftpboot/pxelinux.cfg/defaultファイルで指定したcapall.shやdepall.shスクリプトから呼び出されるスクリプトとなります。

BIOS設定吸い出し用のSSSTK対応のスクリプトcapbios.shを作成し、NFSサーバーの/nfsroot/scriptsディレクトリに置きます。

```
# cd /nfsroot/scripts/  
# vi capbios.sh  
#!/bin/bash  
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt  
cd /TOOLKIT/  
./conrep -s -f /mnt/data_files/bios.cfg  
/bin/umount /mnt
```

スクリプトcapbios.shは、管理対象サーバーがPXEブートした後にSSSTKのconrepコマンドによってBIOS設定がbios.cfgファイルとしてNFSサーバーの/nfsroot/data_filesディレクトリに保存されるようになっています。

BIOS 設定情報を配布する SSSTK のスクリプト作成

BIOS設定配布用のSSSTK対応のスクリプトdepbios.shを作成します。この場合も、管理対象サーバーによって、XMLファイルの指定が必要になる場合がありますので、注意して下さい。

```
# pwd  
/nfsroot/scripts/  
# vi depbios.sh  
#!/bin/bash  
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt  
cd /TOOLKIT/  
./conrep -l -f /mnt/data_files/bios.cfg  
/bin/umount /mnt
```

スクリプトdepbios.shは、管理対象サーバーがPXEブートした後にNFSサーバーの/nfsroot/data_filesディレクトリにあるBIOS設定情報bios.cfgファイルをSSSTKのconrepコマンドでロードすることにより、BIOS設定が自動で行われるようになっています。

[注意] XML ファイルが必要なサーバー機種対応の conrep と SSSTK スクリプト

管理対象のサーバーの機種によってはconrepコマンドに-xオプションでXMLファイルの指定が必要になります。例えば、管理対象サーバーが**HP ProLiant DL170eG6 (HP ProLiant DL2000のシャーシ内のサーバーノード)**の場合、SSSTKを使ったスクリプトcapbios.shやdepbios.shで以下のように指定します。

HP ProLiant DL2000 (DL170eG6) のBIOS設定情報の吸い出し例 :

```
# ./conrep -s -x /mnt/conrep_xml/conrep_DL170eG6_20101217.xml -f /mnt/data_files/bios.cfg
```

HP ProLiant DL2000 (DL170eG6) のBIOS設定情報の配布例 :

```
# ./conrep -l -x /mnt/conrep_xml/conrep_DL170eG6_20101217.xml -f /mnt/data_files/bios.cfg
```

conrepコマンドにおいて、XMLファイルの指定が必要な機種は、SSSTKに含まれるconrep_xmlディレクトリ以

下のXMLファイルのファイル名で分かります。以下はSSSTK 8. 70のXMLファイルが含まれるディレクトリのファイル一覧で表示させたものです。

```
# cd /nfsroot/conrep_xml/  
# ls -l  
CONREP for SL2x170zG6_20100810.xml  
ConrepDL180G6-DL160seG6_20110124.xml  
ConrepSL160sG6_20101201.xml  
DL120G6_20091030_CONREP.XML  
ML110G6_20091030_CONREP.XML  
Readme_CONREP.txt  
conrepDL160G6-SL160zG6_20100723.xml  
conrepDL165G7-SL165zG7-SL165sG7_20110126.xml  
conrep_20100702_SL170s.xml  
conrep_20100920_SL170sG6.xml  
conrep_DL120g5_20080922.xml  
conrep_DL160g5_20081010.xml  
conrep_DL165G5_20100226.xml  
conrep_DL165G5p_20091124.xml  
conrep_DL165G6_20100226.xml  
conrep_DL170eG6_20101217.xml  
conrep_DL170hg6-SL170zg6_20091125.xml  
conrep_DL180g5_20100204.xml  
conrep_DL185G5_20100226.xml  
conrep_ML115g5_20090307.xml  
conrep_ML150g6_20090706.xml  
conrep_SL165zG6_20100414.xml  
conrep_SL170sG6_20110125.xml  
conrepdl160g5p_20081212.xml
```



図 2. HP ProLiant DL2000 は2U で4ノードが稼働する高密度サーバー



図 3. HP ProLiant SL6500 は4U で8ノードが稼働する高密度サーバー



図 4. HP ProLiant DL180G6 は2U で内蔵ディスク 42TB の容量を持つ

上記のXMLのファイル名に記されている機種（DL1000/DL2000シリーズ、SL6000/SL6500シリーズ、DL100シリーズ、ML100シリーズ）はconrepコマンド使用時にXMLファイルの指定が必要ですので注意して下さい。

[注意] 管理対象サーバー機種がXML ファイルを必要とする場合の conrep の入手と配備
管理対象サーバーが、DL1000/DL2000シリーズ、SL6000/SL6500シリーズ、DL100シリーズ、ML100シリーズの場合、conrepコマンドに -xオプションを指定して、機種固有の XML ファイルを指定しての BIOS 設定の取得が必要になります。conrepコマンドに渡すXMLファイルをHPのダウンロードサイトから入手します。以下では、管理対象サーバーが、HP ProLiant DL2000 (DL170e G6) の場合を例に説明します。HP ProLiant DL2000 (DL170e G6) の場合、conrepコマンドをHPのダウンロードサイトから入手できますが、サーバーの機種を問わず適宜最新のconrepコマンドを入手して下さい。

HP ProLiant DL2000 (DL170e G6) に対応したconrepコマンドの入手先URL：
h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?
lang=en&cc=us&pro
dTypeId=15351&prodSeriesId=4324034&prodNameId=5037746&swEnvOID=4103&swLang=8&mode
=2&taskId=135&swItem=MTX-c227b7268dc84ca6ba0bb4217b

入手したCONREPforHPProLiantDL170eG6Server-Linux-2.00.zipファイルをNFSサーバー上に展開します。

```
# pwd
/root
# unzip CONREPforHPProLiantDL170eG6Server-Linux-2.00.zip
# cd ONREPforHPProLiantDL170eG6Server-Linux-2.00
# ls -lF
合計 156
-rw-r--r--. 1 root root 30024 5月 7 11:22 2010 CONREP for Linux on ProLiant 100 User Guide.pdf
-rw-r--r--. 1 root root 2565 5月 7 11:23 2010 Readme_CONREP.txt
-rw-r--r--. 1 root root 103839 11月 12 09:44 2009 conrep
-rw-r--r--. 1 root root 12575 2月 3 10:18 2011 conrep_DL170eG6_20101217.xml
```

conrepコマンドをNFSサーバーの/nfsroot/utilities/ディレクトリにコピーします。

```
# cd /nfsroot/utilities/
# mv conrep conrep_DL360G7
# cp /root/ONREPforHPProLiantDL170eG6Server-Linux-2.00/conrep /nfsroot/utilities/
# chmod 755 conrep
```

以下は、管理対象サーバーがHP ProLiant DL170eG6の場合、SSSTKを使ったスクリプトcapbios.shとdepbios.shです。

capbios.shスクリプト :

```
# vi /nfsroot/scripts/capbios.sh
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/
./conrep -s -x /mnt/conrep_xml/conrep_DL170eG6_20101217.xml -f /mnt/data_files/bios.cfg
/bin/umount /mnt
```

depbios.shスクリプト :

```
# vi /nfsroot/scripts/depbios.sh
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/
./conrep -l -x /mnt/conrep_xml/conrep_DL170eG6_20101217.xml -f /mnt/data_files/bios.cfg
/bin/umount /mnt
```

HP SmartArray 設定情報を吸い出す SSSTK のスクリプト作成

HP SmartArray設定吸い出し用のSSSTK対応のスクリプトcaphpsa.shを作成します。

```
# pwd
/nfsroot/scripts/
# vi caphpsa.sh
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/hpacucli/
./hpacuscripting -c /mnt/data_files/hpsa.cfg -internal
/bin/umount /mnt
```

スクリプトcaphpsa.shは、管理対象サーバーがPXEブートした後にSSSTKのhpacuscriptingコマンドによってSmartArrayのRAID設定がhpsa.cfgファイルとしてNFSサーバーの/nfsroot/data_filesディレクトリに保存されるようになっています。上記の例では「-internal」オプションにより内蔵SmartArray RAIDコントローラーの情報を吸い出すことになります。

HP SmartArray 設定情報を配布する SSSTK のスクリプト作成

HP SmartArray設定配布用のSSSTK対応のスクリプトdephpsa.shを作成します。

```
# pwd
/nfsroot/scripts/
# vi dephpsa.sh
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/hpacucli/
./hpacuscripting -i /mnt/data_files/hpsa.cfg -internal -reset
/bin/umount /mnt
```

スクリプトdephpsa.shは、管理対象サーバーがPXEブートした後にNFSサーバーの/nfsroot/data_filesディレクトリにあるHP SmartArray RAIDコントローラー設定情報hpsa.cfgファイルをSSSTKのhpacuscriptingコマンドでロードすることにより、SmartArray RAIDコントローラーのRAID設定が自動で行われるようになっています。

HP iLO3 設定情報を吸い出す SSSTK のスクリプト作成

iLO3情報吸い出し用のSSSTK対応のスクリプトcapilo3.shを作成します。

```
# pwd
/nfsroot/scripts/
# vi capilo3.sh
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/
/bin/modprobe hpilo
./hponcfg -w /mnt/data_files/ilo3.cfg
/bin/umount /mnt
```

スクリプトcapilo3.shは、管理対象サーバーがPXEブートした後にSSSTKのhponcfgコマンドによってHP ProLiantの遠隔管理チップiLO3の設定がilo3.cfgファイルとしてNFSサーバーの/nfsroot/data_filesディレクトリに保存されるようになっています。

HP iLO3 設定情報を配布する SSSTK のスクリプト作成

iLO3設定配布用のSSSTK対応のスクリプトdepilo3.shを作成します。

```
# pwd
/nfsroot/scripts/
# vi depilo3.sh
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/
/bin/modprobe hpilo
./hponcfg -f /mnt/data_files/ilo3.cfg
/bin/umount /mnt
```

[注意] HP L0100 の設定情報を配布する SSSTK のスクリプト作成

管理対象サーバーがHP ProLiant DL170eG6などの場合、遠隔管理チップにHP Lights-Out 100 (L0100) が採用されています。HP L0100に対応したSSSTKスクリプトcaplo100.shとdeplo100を以下に示します。管理対象サーバーの遠隔感知チップがL0100の場合は、以下のスクリプトを配置して下さい。

```
/nfsroot/scripts/caplo100.shスクリプト :
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/
./lo100cfg -o /mnt/data_files/lo100.cfg
/bin/umount /mnt
```

```
/nfsroot/scripts/deplo100.shスクリプト :
#!/bin/bash
/bin/mount -t $sstk_mount_type -o $sstk_mount_options $sstk_mount /mnt
cd /TOOLKIT/
/bin/modprobe hpilo
./lo100cfg -i /mnt/data_files/lo100.cfg
/bin/umount /mnt
```

BIOS/SmartArray/iLO3 設定情報の吸い出しをバッチ処理するスクリプト作成

上記の吸い出し用スクリプトcapbios.sh、caphpsa.sh、capilo3.shを一斉に実行するバッチ・スクリプトcapall.shを作成します。

```
# pwd
/nfsroot/scripts/
# vi capall.sh
```

```
#!/bin/sh
./capbios.sh
./caphpsa.sh
./capilo3.sh
```

管理対象サーバーにL0100が搭載されている場合は、capilo3.shをcaplo100.shに置き換えてください。

BIOS/SmartArray/iLO3 設定情報の配布をバッチ処理するスクリプト作成

同様に、配布用スクリプトdepbios.sh、dephpsa.sh、depilo3.shを一斉に実行するスクリプトdepall.shを作成します。

```
# pwd
/nfsroot/scripts/
# vi depall.sh
#!/bin/sh
./depbios.sh
./dephpsa.sh
./depi lo3.sh
```

管理対象サーバーにL0100が搭載されている場合は、depilo3.shをdeplo100.shに置き換えてください。上記capall.shとdepall.shは、/var/lib/tftpboot/pxelinux.cfg/defaultファイルに指定したものです。管理対象がPXEブート後、defaultファイルに指定したvmlinuzとinitrd.imgファイルがロードされ、内蔵ディスク等を一切使わずにオンメモリで起動し、上記capall.shスクリプトまたはdepall.shスクリプトがロードされてSSSTKが実行されます。上記全てのスクリプトに実行権限が付与されているかを確認します。付与されていない場合はchmod 755で実行権限を付与します。

```
# pwd
/nfsroot/scripts
# ls -lF
合計 92
-rwxr-xr-x. 1 root root 49 1月 17 01:52 2012 capall.sh*
-rwxr-xr-x. 1 root root 282 1月 17 01:10 2012 capbios.sh*
-rwxr-xr-x. 1 root root 305 1月 17 01:45 2012 caphpsa.sh*
-rwxr-xr-x. 1 root root 299 1月 17 01:51 2012 capilo3.sh*
-rwxr-xr-x. 1 root root 2669 1月 17 00:23 2012 capture.sh*
-rwxr-xr-x. 1 root root 49 1月 17 01:52 2012 depall.sh*
-rwxr-xr-x. 1 root root 282 1月 17 01:42 2012 depbios.sh*
-rwxr-xr-x. 1 root root 312 1月 17 01:45 2012 dephpsa.sh*
-rwxr-xr-x. 1 root root 299 1月 17 01:51 2012 depilo3.sh*
-rwxr-xr-x. 1 root root 2365 3月 10 12:39 2011 deploy.sh*
-rwxr-xr-x. 1 root root 2086 3月 10 12:39 2011 firmware_update.sh*
-rw-r--r--. 1 root root 8329 4月 6 12:42 2010 includes
-rwxr-xr-x. 1 root root 5040 3月 10 12:39 2011 install_centos5.sh*
-rwxr-xr-x. 1 root root 5506 3月 10 12:39 2011 install_rhel4.sh*
-rwxr-xr-x. 1 root root 6548 3月 10 12:39 2011 install_win.sh*
-rwxr-xr-x. 1 root root 5858 3月 10 12:39 2011 install_win2k8.sh*
-rwxr-xr-x. 1 root root 28 3月 10 12:39 2011 shell.sh
```

BIOS、HP SmartArray RAID コントローラー、iLO3 の設定情報の吸い出し

NFSサーバーと同一LANセグメントに配置した管理対象サーバーの電源を投入し、管理対象のサーバーのネットワークカードを使ったPXEブートを行います。HP ProLiantサーバーは、POST画面でF12キーを押すことでPXEブートが可能です。F12キーでのPXE設定は一度だけPXE起動(one-time PXE)するものですので、設定の吸い上げ後に何度もPXEが繰り返される事はありません。PXEブート後、SSSTKのcapall.shスクリプトがロードされ、NFSサーバーの/nfsroot/data_filesディレクトリにbios.cfg、hpsa.cfg、ilo3.cfgファイルが自動的にアップロードされるはずです。SSSTKの実行が進んでいる様子は管理対象サーバーのiLO3の画面で確認できます。

```
ProLiant - 172.16.36.7
Power Switch  Virtual Drives  Keyboard  Help

CLIENT MAC ADDR: 3C 4A 92 E9 18 AC  GUID: 35373932 3438 434E 3731 3033304B5354
CLIENT IP: 172.16.0.9  MASK: 255.255.0.0  DHCP IP: 172.16.1.85
GATEWAY IP: 172.16.1.85

PXELINUX 3.10 2005-08-24  Copyright (C) 1994-2005 H. Peter Anvin
UNDI data segment at:  000942A0
UNDI data segment size: 5C78
UNDI code segment at:  00099F18
UNDI code segment size: 532C
PXE entry point found (we hope) at 99F1:00D6
My IP address seems to be AC100009 172.16.0.9
ip=172.16.0.9:172.16.1.85:172.16.1.85:255.255.0.0
TFTP prefix:
Trying to load: pxelinux.cfg/01-3c-4a-92-e9-18-ac
Trying to load: pxelinux.cfg/AC100009
Trying to load: pxelinux.cfg/AC100008
Trying to load: pxelinux.cfg/AC100007
Trying to load: pxelinux.cfg/AC100006
Trying to load: pxelinux.cfg/AC100005
Trying to load: pxelinux.cfg/AC100004
Trying to load: pxelinux.cfg/AC100003
Trying to load: pxelinux.cfg/AC100002
Trying to load: pxelinux.cfg/AC100001
Trying to load: pxelinux.cfg/AC
Trying to load: pxelinux.cfg/A
Trying to load: pxelinux.cfg/default
boot: _
```

図 5. 管理対象サーバーのPXEブートの様子

```
ProLiant - 172.16.36.7
Power Switch  Virtual Drives  Keyboard  Help

[  1.772503] acpi PNP0303:00: hash matches
[  1.772674] Freeing unused kernel memory: 964k freed
[  1.772989] Write protecting the kernel read-only data: 8192k
[  1.773099] Freeing unused kernel memory: 372k freed
[  1.773309] Freeing unused kernel memory: 1468k freed
INIT: version 2.86 booting

Starting udevd:           done
Loading drivers, configuring devices:   done
Loading drivers...done.
*** Bootstrapping SmartStart Scripting Toolkit ***
*** Detecting system type ***

System Information:
  Name       : ProLiant DL360 G7
  Serial    : CN71030KST
  Arch      : x86_64
  LNAME     : ProLiantDL360G7
  SNAME     : DL360 G7
  Asset Tag :
Not Specified
*** Loading Network Drivers ***
*** Configuring Network Devices ***
-Attempting to configure eth0 using DHCP
```

図 6. 管理対象サーバー上でSSSTKがオンメモリで実行されている様子

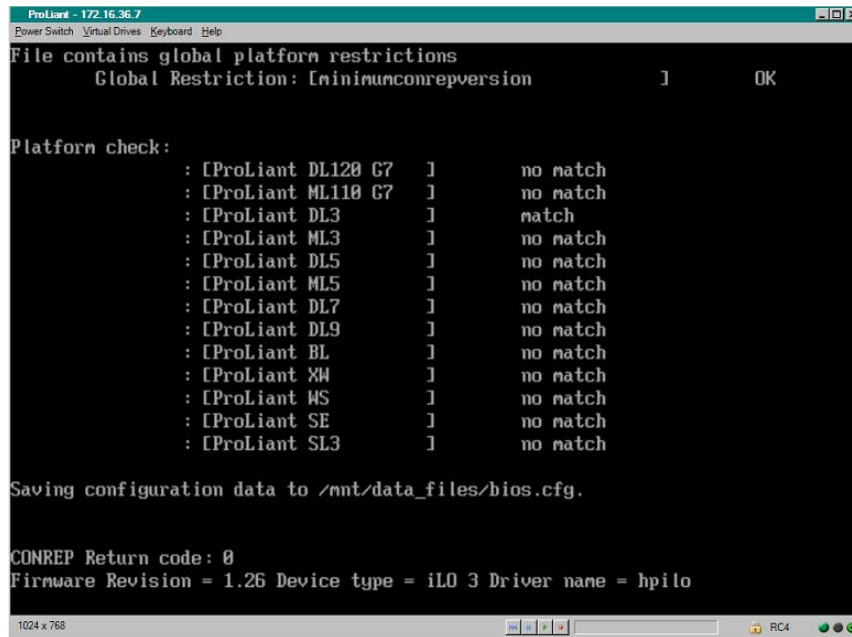


図 7. 管理対象サーバー上でオンメモリで実行されているSSSTKがNFSサーバーの /nfsroot/data_filesディレクトリにbios.cfgファイルをアップロードしていることがわかる

SSSTKによってBIOS、RAIDコントローラー、iLO3の設定情報の吸い出しができたかを確認します。

```
# cd /nfsroot/data_files/
# ls -lF
合計 24
-rw-r--r--. 1 root root 13606 1月 17 02:08 2012 bios.cfg
-rw-r--r--. 1 root root 820 1月 17 02:08 2012 hpsa.cfg
-rw-r--r--. 1 root root 2095 1月 17 02:09 2012 ilo3.cfg
```

BIOS、HP SmartArray RAID コントローラー、iLO3 の設定情報の配布

取得したBIOS設定情報を他の同機種サーバーに展開します。もし何かBIOSの設定情報を変更したものを配布したい場合は、bios.cfgファイルを適宜編集して、配布します。以下では、Hyper-ThreadingとCPUの仮想化機能をDisableにし、BIOSを日本語表示する設定を配布する例を示します。まず、吸い出したBIOS設定ファイルbios.cfgファイルを編集します。bios.cfgファイルの「[Intel_Hyperthreading](#)」セクションの値をEnabledからDisabledに変更します。CPUの仮想化機能は「[CPU_Virtualization](#)」セクションの値をDisabledに変更します。日本語にするには、「[Language](#)」セクションの値を「[JPNJPNus](#)」にします。

```
# cd /nfsroot/data_files/
# cp bios.cfg bios.cfg.org
# vi bios.cfg
...
<Section name="Language" helptext="Defines current language identifier">JPNJPNus </Section>
...
...
```

```
<Section name="CPU_Virtualization" helptext="When enabled, a Virtual Machine Manager supporting this feature can utilize hardware capabilities provided by the server.">Disabled</Section>
```

...

```
<Section name="Intel_Hyperthreading" helptext="Toggles hyperthreading on Intel based G6 and greater systems">Disabled</Section>
```

...

管理対象サーバーがPXEブート後にSSSTKによってdepall.shがロードされるようにするため、NFS/PXEサーバーの/var/lib/tftpboot/pxelinux.cfg/defaultファイルの「default ssttk870cap」をコメントアウトし、「default ssttk870dep」のコメントアウトの記号「#」を外します。

```
# cd /var/lib/tftpboot/pxelinux.cfg/
```

```
# vi default
```

```
#default ssttk870cap
```

```
default ssttk870dep
```

```
prompt 1
```

```
timeout 20
```

```
### SSSTK 8.70 Capture
```

```
label ssttk870cap
```

```
kernel ssttk870/vmlinuz
```

```
append initrd=ssttk870/initrd.img root=/dev/ram0 rw ramdisk_size=190272 ide=nodma ide=noraid
```

```
pnpbios=off network=1 ssttk_mount=172.16.1.85:/nfsroot ssttk_mount_type=nfs
```

```
ssttk_mount_options=rw,nolock ssttk_script=/capall.sh
```

```
### SSSTK 8.70 Deploy
```

```
label ssttk870dep
```

```
kernel ssttk870/vmlinuz
```

```
append initrd=ssttk870/initrd.img root=/dev/ram0 rw ramdisk_size=190272 ide=nodma ide=noraid
```

```
pnpbios=off network=1 ssttk_mount=172.16.1.85:/nfsroot ssttk_mount_type=nfs
```

```
ssttk_mount_options=rw,nolock ssttk_script=/depall.sh
```

これで、管理対象がPXEブート後に、ssttk870depのエントリーが実行され、SSSTKのスクリプトdepall.shがロードされるようになります。BIOS以外のSmart ArrayコントローラーやiLO3の設定の配布を行いたくない場合は、depall.shスクリプトの中のスクリプト名を適宜コメントアウト（「#」記号を行頭に挿入）して下さい。以下の例では、HP SmartArrayとiLO3の設定情報の配布は行わずにBIOSの設定情報のみを配布するdepall.shスクリプトの例です。

```
# cd /nfsroot/scripts/
```

```
# cat depall.sh
```

```
./depbios.sh
```

```
./dephpsa.sh
```

```
./depi lo3.sh
```

再度、管理対象サーバーをPXEブートさせて、SSSTKによってbios.shがロードされているかをiLO3等の遠隔管理画面で確認して下さい。また起動後、bios.shで変更したHyper ThreadingやCPU VirtualizationがdisabledになっているかをBIOS画面等で確認してみてください。

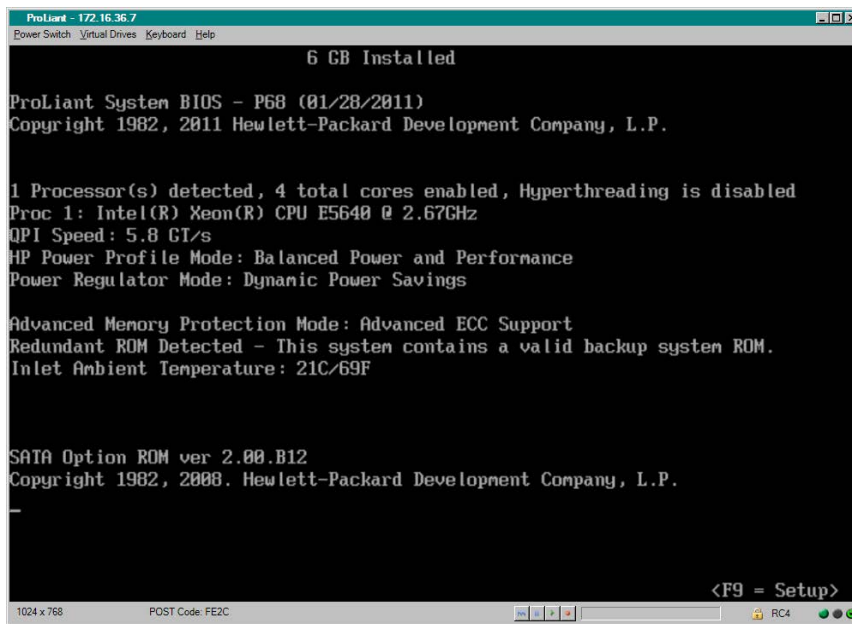


図 8. HP ProLiant DL360G7のPOST画面でHyper-threadingがdisabledになっている旨の表示が見える。 PXE/NFS/SSSTKサーバーによってリモートからBIOSを無人で設定できたことがわかる

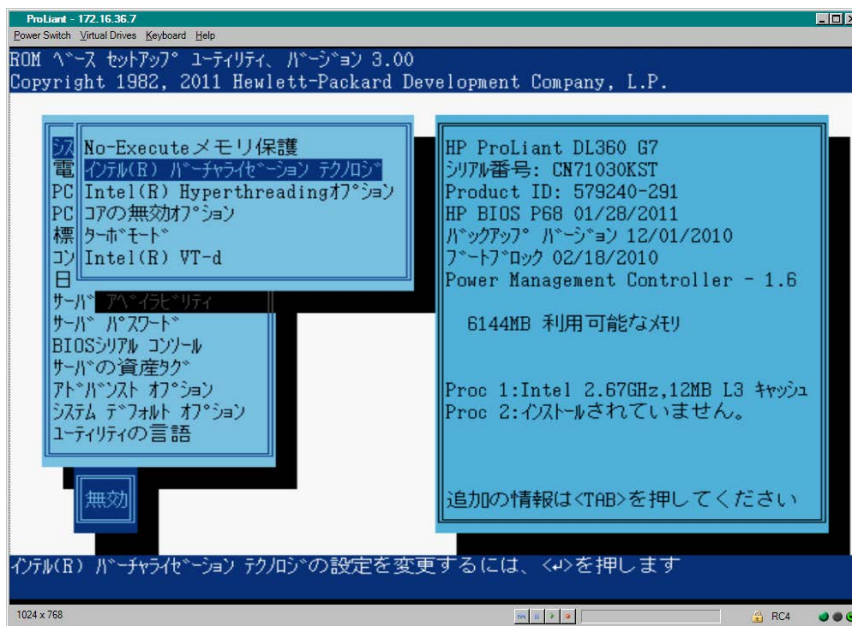


図 9. HP ProLiant DL360G7のBIOS画面でCPUの仮想化機能が「無効」になっている旨の表示が日本語表示されている。 PXE/NFS/SSSTKサーバーによって遠隔からBIOSを無人で設定できたことがわかる

以上