

持続可能な開発のための教育(ESD)に配慮したPC利用 —OpenBSDを利用したファイヤウォールの構築例—

伊藤陽介*

2005年からユネスコによって持続可能な開発のための教育(ESD)が推進されている。情報機器を長期間リユースし続けることによってESDの目標の一つである環境の持続性を保つための価値観と行動の変革の育成をねらい、本論文では、ESDに配慮した具体的な教材例としてセキュリティを確保した上でPCをネットワーク機器の一種であるファイヤウォールとして利用することを提案する。OSを選定した結果、OpenBSDを採用することとし、その構築方法と運用例について述べる。

[キーワード: ESD, OpenBSD, セキュリティ, ファイヤウォール]

1. はじめに

持続可能な開発のための教育(ESD)は2002年に日本政府などが提唱し、2005年にユネスコがESDの主導機関となり文部科学省もESDを推進している[1]。ESDを推進するため「持続可能な社会の創り手の育成」が明記された学習指導要領が2017年に告示された[2~4]。

一方、国税庁は電子計算機のうちサーバ用を除くパーソナルコンピュータ(PC)の耐用年数を4年、その他のもの(PC本体と分離されたディスプレイやプリンタなど)の耐用年数を5年と規定している[5]。そのため賃貸契約で構築された情報システムは、4年または5年が更新することが多い。一方、MM総研によれば2019年のPCの日本国内の出荷台数は1,570,0千台であると報告されている[6]。(一社)パソコン3R推進協会によれば2019年度に回収されたPCは、372.3千台であり、その内リユース台数は18.9千台(5.1%)と報告されている[7]。同年の出荷台数に対するリユース台数の割合は2.4%と著しく低い。回収されたPCの再資源化処理の割合は高いものの、情報化社会の進展と高度化によって使用される情報機器は増加する一方であり、それらを製造したり再資源化処理する場合に発生したりする環境負荷は増大している。

以上述べた状況などを考慮し、耐用年数を超えたり、利用ソフトウェアのサポートが切れたりしたことで廃棄するのではなく、長期間リユースし続けることによってESDの目標の一つである環境の持続性

を保つための価値観と行動の変革の育成をねらう。本論文では、ESDに配慮した具体的な教材例として、PCを長期間に渡ってセキュリティを確保した上でネットワーク機器の一種であるファイヤウォールとして利用することを提案し、その構築方法と運用例について述べる。

2. 教材例

2.1 ハードウェアの選定

本研究室で管理しているPCのうち、正常に稼働しているものの中で最も調達時期の早いPCを2台選定した。以下、選定したPCをPC1及びPC2とする。各PCの主な仕様を表1に示す。両PCとも1998年に調達後、現時点までに22年間経過している。そのため、経年劣化の著しいFDD、HDD及び光学式ドライブは当初の部品から交換されている。また、メモリモジュールも調達時から交換、増設されている。

両PCをネットワーク機器として利用するため、1000Base-Tに対応したネットワーク・インターフェースカード(NIC)を2枚PCIバスに装着している。

2.2 ソフトウェアの選定

PCを安全なネットワーク機器として稼働させるためには、長期間のサポートが期待できるソフトウェアを選定し利用する必要がある。

2.2.1 OS

PC1及びPC2で動作する可能性のあるOSとしてLinux及びBSD系Unixを候補とした。

物理アドレス拡張(PAE)に対応していないインテル系32bit CPUで動作するLinuxディストリビューションとして、Debian 10.4, ubuntu 18.04LTS, Linux

* 鳴門教育大学大学院 高度学校教育実践専攻 自然・生活系教科実践高度化コース(技術・工業・情報科教育実践分野)

表 1 選定したハードウェアの主な仕様

	PC1	PC2
製品名	DELL System Workstation 400 M	DELL System OptiPlex GXi 5200M
ケース	デスクトップ (W419×H165×D445mm)	デスクトップ (W419×H165×D445mm)
マザーボード	チップセット: Intel i440FX	チップセット: Intel i430HX
CPU	PentiumII 266MHz(32bit, L1 Cache 16KB, L2 Cache 512KB), CPU Family: 6, Slot 1(242ピン) 物理アドレス拡張(PAE): 非対応	Pentium MMX 200MHz(32bit, L1 Cache 16KB), CPU Family: 5, Socket 7(321ピン) 物理アドレス拡張(PAE): 非対応
メモリ*	スロット数: 4 規格: EDO DIMM ECC Unbuffered 168pin(3.3V) 実装容量: 384M バイト(最大: 512M バイト)	スロット数: 4 規格: EDO DIMM ECC Unbuffered 168pin(3.3V) 実装容量: 256M バイト(最大: 512M バイト)
FDD*	3.5 インチ, 1.44MB	3.5 インチ, 1.44MB
HDD*	HGST HDT722525DLAT80 (250GB, Ultra ATA133)	HGST HDT722525DLAT80 (250GB, Ultra ATA133)
光学式ドライブ*	MATSHITA SW-9583A(CD-R/RW, DVD-R/RW など)	HL-DT-ST GSA-4167B(CD-R/RW, DVD-R/RW など)
拡張カード* (PCI バス)	ビデオカード: Matrox Electronics Systems MGA 2164W Millennium II NIC: Intel 82540EM (1000Base-T) ×2 USB カード: USB2.0 (3ポート)	NIC: Intel 82540EM (1000Base-T) ×2
内蔵ビデオカード	なし	S3 Trio 64V+ 86c765
内蔵 NIC	3Com 3c905 (100Base-TX)	なし
USB	USB1.0 ×2	USB1.0 ×2
電源ユニット	230W	200W
発売時期	1998 年	1998 年

*: 製品調達時と異なる部品を示す。FDD: フロッピーディスクドライブ, HDD: ハードディスクドライブ

Minto 19.3 が提供され、各サポート期限は 2023 年 4 月とされている。しかし、各 Linux ディストリビューションのインストーラを書き込んだ DVD メディアを用いて PC1 にインストールを試みたが、インストーラ起動直後にカーネルパニックとなり正常に動作しないことが判明した。

つぎに、オープンソースかつ無償提供されている BSD 系 Unix として 32 ビット版の FreeBSD[8], NetBSD[9], OpenBSD[10]を候補とした。FreeBSD と NetBSD はそれぞれ利用のしやすさ、移植性などを目標に開発されている。OpenBSD は NetBSD から分岐して開発された経緯があり、特にセキュリティを重視している点に特徴がある。FreeBSD と NetBSD のサポート期間は約 5 年間である。一方、表 2 に示すように、最近リリースされた OpenBSD のサポート期間は約 1 年間と短い、約半年毎にリリースされるバージョンに更新を繰り返すことでサポート期間を延長し続けることができる。

FreeBSD 12.1 のインストーラを書き込んだ DVD メディアを用いて PC1 にインストールすることを試みたが、インストーラ起動直後にカーネルパニックとなり正常に動作しないことが判明した。

また、NetBSD のメーリングリスト tech-kern によれば、OpenBSD のプロジェクトの一環として開発された堅牢性の高いネットワーク・パケット制御用ソフトウェアの一種である Packet Filter(pf)[11]のサポートを打ち切る内容の投稿が 2019 年 3 月にあった

表 2 OpenBSD のバージョン履歴(抜粋)

バージョン	リリース日	サポート期限
6.4	2018 年 10 月 18 日	2019 年 10 月 17 日
6.5	2019 年 4 月 24 日	2020 年 5 月 19 日
6.6	2019 年 10 月 17 日	2020 年 10 月 18 日
6.7	2020 年 5 月 19 日	2021 年 5 月
6.8	2020 年 10 月 18 日	2021 年 10 月

[12]。この理由により NetBSD を採用しないこととした。

OpenBSD 6.7 のインストーラを書き込んだ DVD メディアを用いて PC1 にインストールすることを試みたが、インストーラ起動直後にカーネルパニックとなり正常に動作しないことが判明した。Linux や FreeBSD と異なり、OpenBSD ではフロッピーディスク用インストーラも提供されているため、それを用いたところ PC1 及び PC2 でインストーラを正常に起動させることに成功した。インストーラ起動後、OpenBSD のファイルセット配布用サーバからネットワーク経由でインストールに必要なファイルを読み込むことで完了した。

以上述べた結果から、PC1 及び PC2 で動作可能な OS として OpenBSD を採用し、関連するソフトウェアを利用することとした。

2.2.2 ネットワーク・パケット制御用ソフトウェア

OpenBSD ではネットワーク・パケット制御用ソフトウェアとして pf を利用する。pf はパケットフィル

タリングやネットワーク・アドレス変換(NAT)を行う機能などを備えている。

3. OpenBSD を利用したファイアウォール

第2章で述べたPCを対象としてOpenBSDを利用したファイアウォールの構築と運用方法について述べる。

3.1 インストール方法

2.2.1節で述べたようにOpenBSDはPC1及びPC2にインストールできた。以下に、OpenBSD 6.7のインストール方法について述べる。

3.1.1 インストーラの準備

OpenBSDのWebページ<https://www.openbsd.org/faq/faq4.html#Download>にある「floppyXX.fs」の「i386」を選択し、フロッピーディスク用インストーラのイメージファイル「floppy67.fs」をダウンロードする。ここで、「67」は、OpenBSDのバージョン「6.7」を意味する。なお、ミラーサイト(例えば、<http://ftp.riken.jp/pub/OpenBSD/6.7/i386/>)からイメージファイルをダウンロードしてもよい。

インストーラの起動時に用いるフロッピーディスクは、FDDを備えたLinuxマシンを使って「floppy67.fs」を

```
dd if=floppy67.fs of=/dev/fd0 bs=1024 conv=sync; sync
```

としてフロッピーディスクに書き込む。ここで、「/dev/fd0」はFDDのデバイス名である。

3.1.2 インストール手順

PCIバスに2枚のNICを装着しておき、インターネット接続できdhcpサービスを受けられるグローバル側ネットワークのNICにLANケーブルを装着しておく。ここでは、Intel製NICのドライバ(em)を用いた場合について述べる。以下、□で囲んだ文字は表示結果を示す。網掛け文字は説明を表し実際に表示される内容とは異なる。

3.1.1節で述べた方法で準備したフロッピーディスクを用いてインストーラを起動する。

```
Welcome to the OpenBSD/i386 6.7 installation program.
(I)nstall, (U)pgrade, (A)utoinstall or (S)hell?
```

「I」を入力し、インストールを開始する。

```
Choose your keyboard layout ('?' or 'L' for list) [default]
```

キーボードのレイアウト選択を日本語とするため「jp」を入力する。

```
System hostname? (short form, e.g. 'f
```

```
oo')
```

当該PCにつけるホスト名を入力する。

```
Available network interfaces are: em0
em1 vlan0.
Which network interface do you wish to
configure? (or 'done')[em0]
```

em0とem1の2つのNICが認識されている。デバイス名はNICの種類によって異なる場合がある。ここでは、ネットワークインストールに用いるNICとしてem0を選択する。なお、em0がインターネットに接続されていない場合、em1を選択する。

```
IPv4 address for em0? (or 'dhcp' or
'none') [dhcp]
em0: IPv4アドレス lease accepted from
dhcpサーバのIPv4とMACアドレス
IPv6 address for em0? (or 'autoconf'
or 'none') [none]
```

選択したNICのIPv4アドレスはdhcpサーバを使って設定させ、IPv6は使用しない。

```
Available network interface are: em0,
em1 vlan0
Which network interface do you wish to
configure? (or 'done') [done]
Using DNS domainname ドメイン名
Using DNS nameservers DNSサーバのIPv4
アドレス
Password for root account? (will not
echo)
```

他のNICを設定する必要がないので、doneを選択する。dhcpサーバから提供されたドメイン名とDNSサーバの情報が表示される。rootアカウントのパスワードを入力する。

```
Start sshd(8) by default? [yes]
Do you expect to run the X Window Sys-
tem? [yes]
Do you want the X Window System to be
started by xenodm(1)? [no]
Setup a user? (enter a lower-case log
inname, or 'no') [no]
Since no user was setup, root logins
via sshd(8) might be usefull.
WARNING: root is targeted by password
guessing attacks, pubkeys are safer.
Allow root ssh login? (yes, no, prohi-
bit-password) [no]
```

sshdの起動、X Windowはインストールするが起動時は利用しない。一般ユーザの設定なし、sshによるrootログインを不許可とする。なお、一般ユーザの設定についてはインストール完了後に行う。

```
Available disks are: wd0.
Which disk is the root disk? ('?' for
details) [wd0]
```

利用可能なディスクとしてハードディスクのデバイス名wd0が表示される。インストール対象とするデバイス名をwd0とする。

```
Use (W)hole disk, use the (O)penBSD a
```

```
rea or (E)dit the MBR? [OpenBSD]w
wd0 のパーティション情報が表示される。今回は
すべての領域を使用するため「w」を選択する。
```

```
Use (A)uto layout, (E)dit auto layout,
or create (C)ustom layout? [a]
```

自動でパーティションが作成された後、各パーティションがフォーマットされる。

```
Which disk do you wish to initialize?
(or 'done') [done]
```

さらに初期化するディスクはないため、そのままエンターキーを押す。

```
Let's install the sets!
Location of sets? (cd0 disk http or
'done') [cd0] http
HTTP proxy URL? (e.g. 'http://proxy:8080',
or 'none') [none] プロキシのURL
HTTP Server? (hostname or 'done') ミ
ラーサーバのホスト名
Server directory? [pub/OpenBSD/6.7/i386]
```

ネットワークインストールするため「http」を選択し、必要に応じてプロキシの URL を入力し、ミラーサーバのホスト名とディレクトリを設定する。日本国内のミラーサーバのホスト名として、ftp.jaist.ac.jp や ftp.riken.jp などが利用できる。

```
Select sets by entering a set name, a
file name pattern or 'all'. De-select
sets by prepending a '-', e.g.: '-game*'.
Selected sets are labelled
'[X]'.
[X] bsd [X] comp67.tgz [X] x
base67.tgz [X] xserv67.tgz
[X] bsd.rd [X] man67.tgz [X] x
share67.tgz
[X] base67.tgz [X] game67.tgz [X] x
font67.tgz
Set name(s)? (or 'abort' or 'done')
[done]
```

ミラーサーバと通信後、ファイルセットの一覧が表示される。すべてのファイルセットをインストールする。

```
Get/Verify SHA256.sig . . .
Signature Verified
Get/Verify bsd . . .
Get/Verify bsd.rd . . .
Get/Verify base67.tgz . . .
Get/Verify man67.tgz . . .
Get/Verify game67.tgz . . .
Get/Verify xbase67.tgz . . .
Get/Verify xshare67.tgz . . .
Get/Verify xfont67.tgz . . .
Get/Verify xserv67.tgz . . .
Installing bsd . . .
Installing bsd.rd . . .
Installing base67.tgz . . .
Extracting etc.tgz . . .
```

```
Installing comp67.tgz . . .
Installing man67.tgz . . .
Installing game67.tgz . . .
Installing xbase67.tgz . . .
Installing xetc.tgz . . .
Installing xshare67.tgz . . .
Extracting xetc.tgz . . .
Installing xshare67.tgz . . .
Installing xfont67.tgz . . .
Installing xserv67.tgz . . .
```

ミラーサーバからファイルセットがダウンロードされた後、インストールされる。なお、「・・・」の部分に進捗状況を示す文字列とファイルサイズなどが表示される。

```
What timezone are you in? ('?' for list)
[Canada/Mountain] Japan
```

タイムゾーンは日本とするため「Japan」と入力する。

```
Saving configuration files... done.
Making all device nodes... done.
Relinking to create unique kernel...
done.
CONGRATULATIONS! Your OpenBSD install
has been successfully completed!
When you login to your new system the
first time, please read your mail using
the 'mail' command.
Exit to (S)hell, (H)alt or (R)eboot?
[reboot]
```

インストールが完了後、フロッピーディスクを取り外し再起動させる。

再起動後、root でログインし、ネットワークの設定を行う。

```
/root/.profile
```

に以下の環境変数を設定するスクリプトを追加する。

```
PROXYSV="プロキシのURL"
export http_proxy=$PROXYSV
export https_proxy=$PROXYSV
export HTTP_PROXY=$PROXYSV
export HTTPS_PROXY=$PROXYSV
```

デバイス名 em0 の NIC に dhcp サーバから割り当てられるネットワーク情報を設定する場合、

```
/etc/hostname.em0
```

というファイルに

```
dhcp
```

と記述する。

デバイス名 em1 の NIC に、IPv4 アドレスとネットマスクを設定する場合、

```
/etc/hostname.em1
```

に

```
inet IPv4 アドレス ネットマスク
```

を記述する

ゲートウェイの IPv4 アドレスは、

/etc/mygate
に記述する。

3.1.3 パッチの適用とパッケージの更新

インストールした OS に最新のパッチを適用し、
パッケージを更新するため

```
syspatch  
pkg_add -u
```

を実行した後、再起動する。

3.2 運用手順

3.2.1 OS のバージョンアップ

OS のバージョンアップを行う前に、最新のパッチ
を適用し、各パッケージを更新しておく。

バージョンアップは、
sysupgrade
を実行することで、/etc/installurl に記述されて
いるミラーサーバから必要なファイルがダウンロード
され、自動的にインストールが行われる。さらに、
カーネルの再リンク後、再起動がかかる。

3.2.2 OS の保守

定期的に OS の保守を行うため、3.1.3 節で述べた
パッチの適用とパッケージの更新を行う。

3.3 ファイヤーウォールの構築

2.2.2 節で述べたネットワーク・パケット制御用
ソフトウェア pf を用いて、ファイヤーウォールを構築
する。

ここで、em0 をグローバル側ネットワーク、em1 を
ローカル側ネットワークとする。まず、IPv4 パケッ
トのフォワーディングを有効にするため、

```
/etc/sysctl.conf  
に  
net.inet.ip.forwarding=1
```

を追記する。

パケットフィルタリングや IP マスカレードのルー
ルは、/etc/pf.conf に記述する。このファイルは起
動時に読み込まれ反映される。

起動後、/etc/pf.conf を編集した場合
pfctl -f /etc/pf.conf
を実行して反映させることもできる。なお、ルール
の文法チェックのみの場合
pfctl -nf /etc/pf.conf
とする。

pf は、以下に示す行単位で記述されたルールを始
めから順番に評価する。

```
action [direction] [log] [quick] [on interface]
```

```
[af] [proto protocol] [from src_addr [port src  
_port]] [to dst_addr [port dst_port]] [flags tc  
p_flags] [state]
```

各ルールの記述方法については、参考文献[11]に
記載されているが、その一部を和訳し整理したもの
を表 3 に示す。

pf ではマッチするルールがあってもそこで評価は
停止せず、最後にマッチしたルールが適用される。
ただし、quick をつけた場合にはそこで評価が終了
し、そのルールが適用される。どのルールにもマッ
チしないパケットは、すべて通過してしまうので、
そうならないようにルールを記述しなければならない。

pf のログは、/var/log/pflog に書き込まれるが、
テキストファイルでないため、
tcpdump -n -e -ttt -r /var/log/pflog
として表示する。また、リアルタイムでログを表示
する場合

```
tcpdump -n -e -ttt -i pflog0
```

を実行する。

例えば、ローカル側の em1 からグローバル側の em0
に、IP マスカレードして通信するルールは、以下のよ
うに記述する。ここで「¥」は継続行を示す。

```
set skip on lo  
block return  
pass  
match out on em0 inet ¥  
from ローカル側のネットワークアドレス ¥  
nat-to (em0)
```

さらに、許可する tcp 及び udp のポート番号を考慮
したルール例を表 4 に示す。グローバル側及びローカ
ル側の各 NIC に対応するデバイス名、ローカル側ネッ
トワークアドレス、許可する tcp 及び udp のポート番
号をサービス名または数値の集合をそれぞれ変数に代
入し、各ルールの記述時に用いる。ルールの役割は表
4 内にコメントで記載している。

また、X Window を startx コマンドで起動後、ネッ
トワークプロトコルアナライザである wireshark[13]
を利用すると、より細かいネットワーク・パケットの
状態も把握できる。

3.4 ファイヤーウォールの運用例

PC1 と PC2 に OpenBSD 6.7 をインストールし、表 4 に
示したルールを用いたファイヤーウォールを 2020 年 5 月
に構築し、24 時間稼働を開始した。当初、Realtek 製
NIC を利用していたところ「watchdog timeout」が頻発
したため、すべて Intel 製 NIC に交換した。その後、
OpenBSD 6.8 にバージョンアップしたが、問題なく稼
働している。

PC1 と PC2 に iperf3[14]をインストールし、通信速度を実測したところ、それぞれ 80Mbps, 31Mbps 程度であった。32 ビット PCI バスに装着した 1000Base-T 対応の NIC が備える性能と比較するとかなり処理能力は劣るが、インターネット利用であれば実用上問題のないレベルで通信できると判断している。

4. まとめ

本論文では、ESD に配慮した具体的な教材例として、約 22 年前に調達した PC に OpenBSD をインストールし、長期間に渡ってセキュリティを確保した上でファイアウォールとして実際に利用できることを示した。

FDD やフロッピーディスク、IDE 接続のハードディ

表 3 pf によるルール設定 (抜粋)

文法	設定内容
action (必須)	通信を許可するときは「pass」、通信を許可しパラメータを設定するときは「match」、拒否するときは「block」と記述する。「match」の場合、最後に一致したルールではなく、ルールが一致するたびにパラメータが設定される。「match」は、nat-to, binat-to, rdr-to, scrub とともに使う。「block」の場合、さらにオプションをつけて「block drop」とすると、何も返信せず拒否し、「block return」とすると、TCP であれば TCP RST パケットを返信し、ICMP であれば ICMP Unreachable パケットを返信する。
direction	入ってくる通信に対するルールのときは「in」、出ていく通信に対するルールのときは「out」と記述する。何も記述しなかった場合、入ってくる通信と出ていく通信の両方に対するルールとなる。
log	ログをとる場合「log」と記述する。
quick	当該ルール以降に記述されたルールを適用しない場合「quick」と記述する。すなわち、この時点でルールの評価が終了し、当該ルールが適用される。
on interface	ルールを適用するネットワークインターフェースのデバイス名またはグループ名を「on」の次に記述する。グループ名の「any」を使って、「on any」と記述すると、すべてのネットワークインターフェースのデバイスが対象となる。
af	address family の略である。IPv4 のみに適用する場合「inet」と記述し、IPv6 のみに適用する場合「inet6」と記述する。何も記述しない場合、IP アドレスの指定について限定することはない。
proto protocol	ルールを適用するプロトコルを記述する。プロトコルとして、tcp, udp, icmp, icmp6 などがある。
from src_addr [port src_port]] [to dst_addr [port dst_port]]	src_addr は送信元アドレス、dst_addr は送信先アドレスを指定する。単一の IPv4 または IPv6 アドレスを記述したり、IPv4 または IPv6 アドレス群を記述したりできる。ネットワークデバイス名に「:network」を付けると、そのネットワークアドレスとなる。ネットワークデバイス名に「:broadcast」を付けると、そのネットワークのブロードキャストアドレスとなる。すべてのアドレスの場合「any」を記述する。「!」を付けると論理否定を意味する。「all」は「from any to any」と同じ意味になる。 src_port は送信元ポート番号、dst_port は送信先ポート番号を指定する。1~65535 の番号やサービスの名称 (例 ssh) を記述できる。関係演算子「!=, <, >, <=, >=」を使ってポート番号の範囲を示す。
flags tcp_flags	proto tcp を使用する場合に、TCP ヘッダにセットされているべきフラグを指定する。このフラグは、flags check/mask として指定される。
state	パケットがこのルールにマッチした場合、状態の情報を保持するかどうかを指定する。「keep state」と記述すると、TCP, UDP および ICMP に対して有効である。「modulate state」と記述すると、TCP に対してのみ有効となる。「synproxy state」と記述すると、詐称された TCP SYN flood からサーバを防護するため、着信する TCP 接続のプロキシを行う。このオプションは、keep state および modulate state の機能を含んでいる。

表 4 ファイアウォールのルール例

```

ext_if = "em0"           # グローバル側NICのデバイス名
int_if = "em1"          # ローカル側NICのデバイス名
int_net = $int_if:network # ローカル側ネットワークアドレス
services_tcp="{ ssh, auth, ntp, smtp, smtps, imaps, www, https }" # 許可するtcpのポート番号の定義
services_udp="{ domain, ntp, https }" # 許可するudpのポート番号の定義
set skip on lo0        # ループバックインタフェースはフィルター対象外
set block-policy drop  # ブロックしたパケットに対して何も返さない。
match in all scrub (no-df) # パケットの正規化
block in log all       # 全てのパケットをブロック
block out all
block in quick inet6 all # IPv6はすべてブロック
antispoof quick for { lo0, $ext_if, $int_if } # 詐称されたパケットはブロック
match out on $ext_if inet from $int_net nat-to ($ext_if) # IPマスカレード
pass in on $int_if proto tcp from $int_net to any port $services_tcp modulate state
pass out on $ext_if proto tcp from $ext_if to any port $services_tcp modulate state
pass in on $int_if proto udp from $int_net to any port $services_udp modulate state
pass out on $ext_if proto udp from $ext_if to any port $services_udp modulate state
pass in on $int_if proto icmp from $int_net to $int_if keep state # ローカル側からのICMPを通す。

```

スクなど、現在では入手困難な部品があり、新規に OpenBSD をインストールすることが難しい状況となっている。将来的にフロッピーディスクを使わなくても OpenBSD をインストールできるように、今回 OpenBSD を構築した HDD のクローンやイメージファイルを作成しておく必要がある。

レアメタルなど様々な資源と多大なエネルギーを利用して製造された情報機器を長期間リユースし続けるという取り組みは、持続可能な社会を目指す一歩になると考えられる。ESD の目標の一つである環境の持続性を保つための価値観と行動の変革を育成するために、今後は PC のファイヤウォール以外の利用時の教材化についても模索していく予定である。

参考文献

- [1] 文部科学省(2018) ESD(持続可能な開発のための教育)推進の手引, 国際統括官付, 日本ユネスコ国内委員会, https://www.mext.go.jp/unesco/004/_icsFiles/afieldfile/2018/07/05/1405507_01_2.pdf (最終アクセス日: 2021年2月1日).
- [2] 文部科学省(2017) 小学校学習指導要領(平成29年告示), https://www.mext.go.jp/content/1413522_001.pdf (最終アクセス日: 2021年2月1日).
- [3] 文部科学省(2017) 中学校学習指導要領(平成29年告示), https://www.mext.go.jp/content/1413522_002.pdf (最終アクセス日: 2021年2月1日).
- [4] 文部科学省(2018) 高等学校学習指導要領(平成30年告示), https://www.mext.go.jp/content/1384661_6_1_3.pdf (最終アクセス日: 2021年2月1日).
- [5] 国税庁(2020) 減価償却資産の耐用年数等に関する省令の別表, 別表第一 機械及び装置以外の有形減価償却資産の耐用年数表, 令和二年三月三十一日財務省令第二六号, <https://elaws.e-gov.go.jp/document?lawid=340M50000040015> (最終アクセス日: 2021年1月29日).
- [6] 株式会社MM総研(2020) 2019年暦年国内パソコン出荷概要, <https://www.m2ri.jp/release/detail.html?id=399> (最終アクセス日: 2021年2月1日).
- [7] (一社)パソコン3R推進協会(2020) 2019年度のパソコンメーカーによる使用済パソコンの回収再資源化実績は 372.3 千台, <https://www.pc3r.jp/topics/200804.html> (最終アクセス日: 2021年2月1日).
- [8] The FreeBSD project, <https://www.freebsd.org/> (最終アクセス日: 2021年2月4日).
- [9] The NetBSD project, <https://www.netbsd.org/> (最終アクセス日: 2021年2月4日).
- [10] The OpenBSD project, <https://www.openbsd.org/> (最終アクセス日: 2021年2月4日).
- [11] OpenBSD PF - User's Guide, <https://www.openbsd.org/faq/pf/> (最終アクセス日: 2021年2月4日).
- [12] M. Villard(2019) Removing PF, NetBSD Mailing Lists, tech-kern, <http://mail-index.netbsd.org/tech-kern/2019/03/29/msg024883.html> (最終アクセス日: 2021年2月4日).
- [13] The Wireshark Foundation(2021) Wireshark, <https://www.wireshark.org/> (最終アクセス日: 2021年2月15日).
- [14] ESnet(2020) iperf3, <https://software.es.net/iperf/> (最終アクセス日: 2021年2月15日).