Использование цифровых водяных знаков и небайесовские задачи распознавания

Писковский Виктор Олегович

План

1. Абонентский облачный терминал (Endpoint Cloud Terminal)
2. Цифровой водяной знак для идентификации рабочего места
3. Задачи распознавания

Endpoint Cloud Terminal as an Approach to Secure the Use of an Enterprise Private Cloud

A.Grusho, V. Sentchilo, E. Timonina
Federal Research Center
"Computer Science and Control"
Russian Academy of Sciences

A. Nikolaev
N.N. Semenov Federal Research Center for Chemical Physics
Russian Academy of Sciences

V. Piskovski
The faculty of Computational Mathematics and Cybernetics
Lomonosov Moscow State University

Moscow,
Russia 2020

Purpose and Aims

Application area are departments, organizations and companies working with information, the loss of which can have negative consequences, e.g.:

- ✓ Government and state organizations
- ✓ oil and gas enterprises
- ✓ energy sector
- ✓ manufacturing and construction enterprises
- ✓ banks, financial and insurance companies

Security issues. ***ECT aims to diminish the risk of attacks***

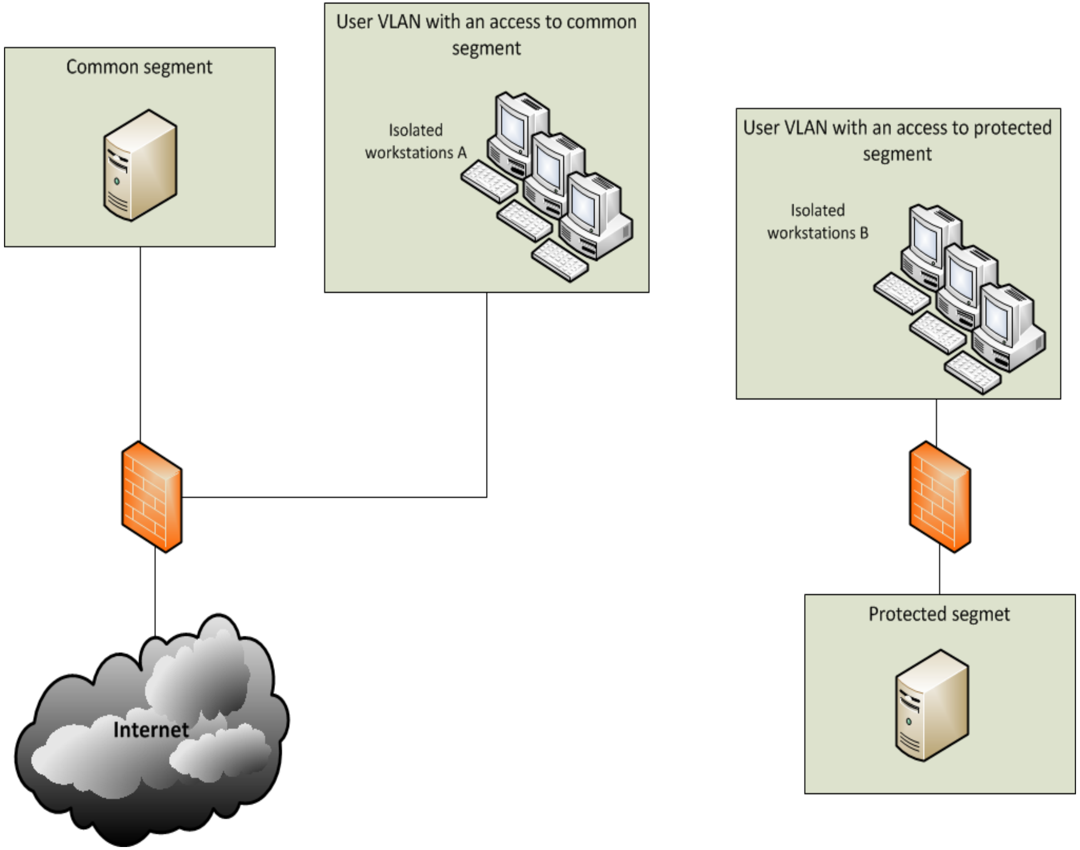
Intruder	Risks
Outsider	Vulnerability exploitation
Insider	Security policy violation, e.g. • BYOD, remote workstations, business trips, … • Invited R&D companies • The loss of a notebook, etc

Practical issues, i.e. optimizing capital and operating costs:

- processing information of various categories on one hardware
- collective use of licensed software
- usage external hard disk3 or storage system
- collective use of pre-configured VM from corporate repositories

Common Practice

Practical activities in organizations that process significant amounts of confidential information usually require providing employees with the ability to simultaneously work with internal, distributed information resources and access to the Internet. The need to ensure information security requires the use of appropriate organizational and technical methods of information protection, which are based on the idea of domain isolation


Common segment
User VLAN with an access to common segment
Isolated workstations A
User VLAN with an access to protected segment
Isolated workstations B
Internet
Protected segmet

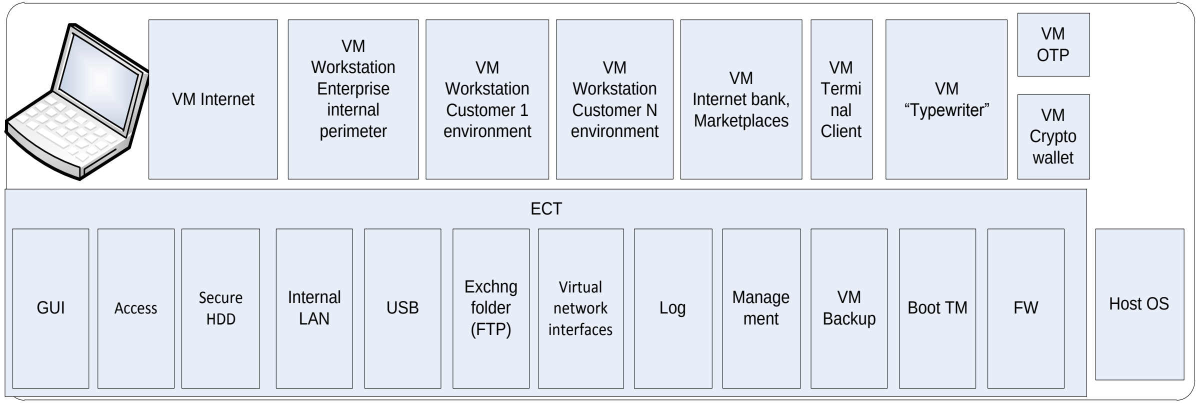
WHAT IS ECT

Endpoint Cloud Terminal (ECT)
General-purpose software with built-in security features

ECT:

- ✓ **Can be deployed on a notebook or a workstation**
- ✓ **Provides secure processing confidential data in isolated environments**

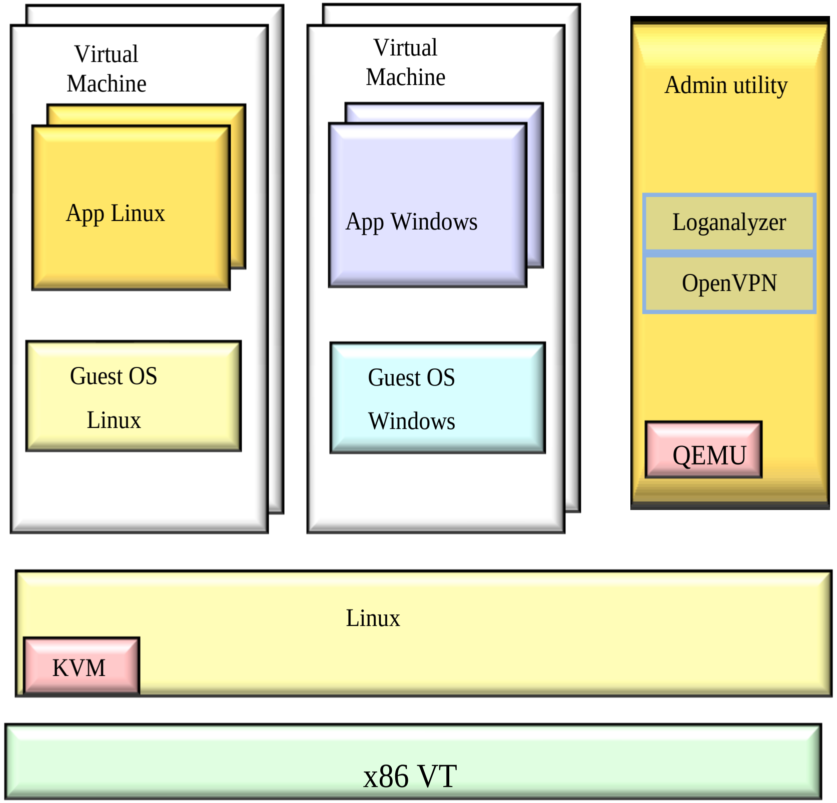
Components	Responsibility
Hosting OS	Running services and functions
ECT	Management and control: VM, peripheries' access, HDD volumes, DHCP, NAT, VPN, FTP, FW, security policies, TM, AV, log, ...
Virtual machines (VM)	Running of isolated workplace environments preconfigured for users duties MSWin, Linux, Ux


VM Internet
VM Workstation Enterprise internal perimeter
VM Workstation Customer 1 environment
VM Workstation Customer N environment
VM Internet bank, Marketplaces
VM Termi nal Client
VM "Typewriter"
VM OTP
VM Crypto wallet
ECT
GUI
Access
Secure HDD
Internal LAN
USB
Exchng folder (FTP)
Virtual network interfaces
Log
Manage ment
VM Backup
Boot TM
FW
Host OS

ARCHITECTURE

The implementation of ECT is based on the use of virtualization technology. The software consists of a KVM module running at the Linux kernel level, as shown in Figure QEMU emulates the hardware of Intel x86 processors and virtual workstation I/O devices.

The Linux kernel's KVM module allows QEMU running in user space to use the hardware virtualization capabilities provided by modern x86-compatible processors to improve performance of virtual workstations.



REQUIREMENTS

- ✓ Availability of a secure channel from the workstation to the entry point of internal network or to the access point of a centralized resource
- ✓ Using the CSP recommended by the regulator
- ✓ Use of certified software and equipment protected from unauthorized access
- ✓ Minimizing the complexity and quantity of equipment used, minimizing the architecture of the information technology tools used, simplifying physical maintenance, remote maintenance and administration
- ✓ Ensuring the possibility of using an electronic signature in the following modes:
  - Signature locally in the workstation
  - Cloud signature
- ✓ Ability to work with all virtual resources of the enterprise, including electronic document management
- ✓ Ability to reuse existing equipment

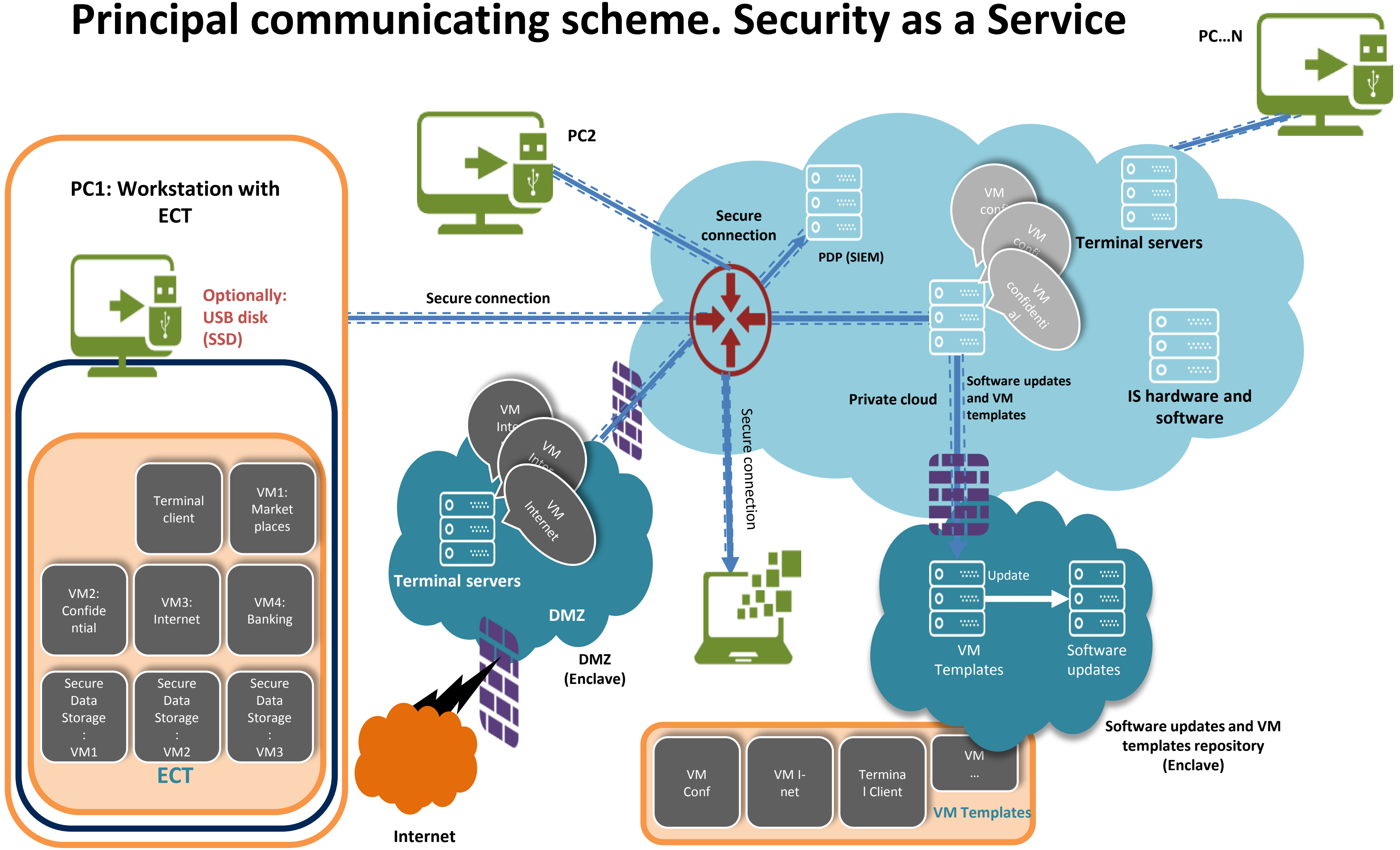
Compensating the following:

- ✓ Risks caused by the necessity of having a reliable broadband network connection, client workstation services centralized (cloud) of the company information infrastructure (hereinafter - CII)
- ✓ The lack of channel encryption in the classic terminal solutions
- ✓ Insufficiently developed issues of ensuring the security of the client's place
- ✓ The fundamental impossibility of rapid pre-processing primary data in the workstation leads to a disproportionate increase in the load of communication lines and dedicated servers for processing this data. The terminal solution hardly help in primary preprocessing such data as well
- ✓ The problem of connecting users to the Internet via CII leads to an increase in the load on communication channels, decentralized, immediate connections means more complicated workstation software and hardware.
- ✓ The necessity to use a separate computer to access the Internet in both cases leads to complexity of the architecture and additional costs
- ✓ The usage of personal computing equipment, BYOD approach, means:

  - untrusted channels and hardware to work with confidential data outside the protected perimeter.

  - risks of loss and violation of data integrity, information leakage.

Principal communicating scheme. Security as a Service



ENTERPRISE. SECURITY AS A SERVICE

The most tough and expensive tasks:

→ Configuration management
→ Change management
→ Release management

Enterprise compounds

Preconfigured VM, Patches and PTF repository

Preconfigured VM, Patches and PTF repository.

The repository for referenced VM containers and updates , standardized and well-tuned VMs with software updates.

Production landscape

The on-duty landscape of CII is a production servers and services including its components, network services, VMs, and all the stuff passed tests procedures and accepted for enterprise operation

DMZ

Secure isolated work in Internet via open channels.

ECT functions:

→ Terminal mode.
→ Secure terminal access to the VDI of the CII.
→ Secure terminal access to the DMZ VDI.
→ Replicate the referenced VM to a personal computer, if needed
→ Replicate the DMZ referenced VM to a personal computer, if needed.
→ Safe simultaneous operation in all possible modes.
→ VPN secure channel.
→ Safe completely isolated off-line mode.

Results

ECT:

- ✓ does not require a mandatory network connection
- ✓ does not impose restrictions on the use of the operating system
- ✓ allows to perform the necessary pre/post processing of data locally to meet the secure requirements
- ✓ provides simultaneous (without rebooting) work on personal computers in various categorized segments of the corporate network, for example, the work of traders with trading platforms and simultaneously in the protected perimeter of CII
- ✓ provides employees with remote workplaces, e.g. from home or on business trips
- ✓ makes available a safe operation on any computer (shared, home) from an individual portable USB drive
- ✓ seamless migration of the VM from CII to the workstation and back if necessary
- ✓ instant switching between VMs as well as CII domains
- ✓ reliable implementation of DAC to data and peripherals at the hypervisor level
- ✓ audit of events and monitoring network interaction
- ✓ decreasing of operational costs due to common usage of enterprise secure measures

The International Science and Technology Conference
«Modern Network Technologies, MoNeTec - 2020»

The Research of a Method to Identify a Workplace via a Monitor Snapshot

A.Grusho[1], V. Piskovski[2], D. Semenikhin[2], I. Sudarikov[1], E. Timonina[1]

1 - Federal Research Center "Computer Science and Control", Russian Academy of Sciences
2 - The faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University
Moscow, Russia

October 27-29, 2020
Digital format (ONLINE)

A problem

- Unauthorized leakage of confidential information is one of the most eminent nowadays threats (e.g. $0.5 per a personal record with such details as id, bank accounts, phone number, etc.)
- One side - counter measures: technical and administrative
- Another side - private gadgets with photo and video recording facilities
- The problem is to identify the employee or his workplaces of the attack, and timestamp
- The finally aim is to make stealing information not worth the effort given

Purpose and Aims

Application area are departments, organizations and companies working with information, the loss of which can have negative consequences, e.g.:

- ✓ Government and state organizations
- ✓ Commercial companies
  High tech companies
- ✓ banks, financial and insurance companies
- ✓ etc

Security issues. ***The product aims to diminish the risk of attacks***

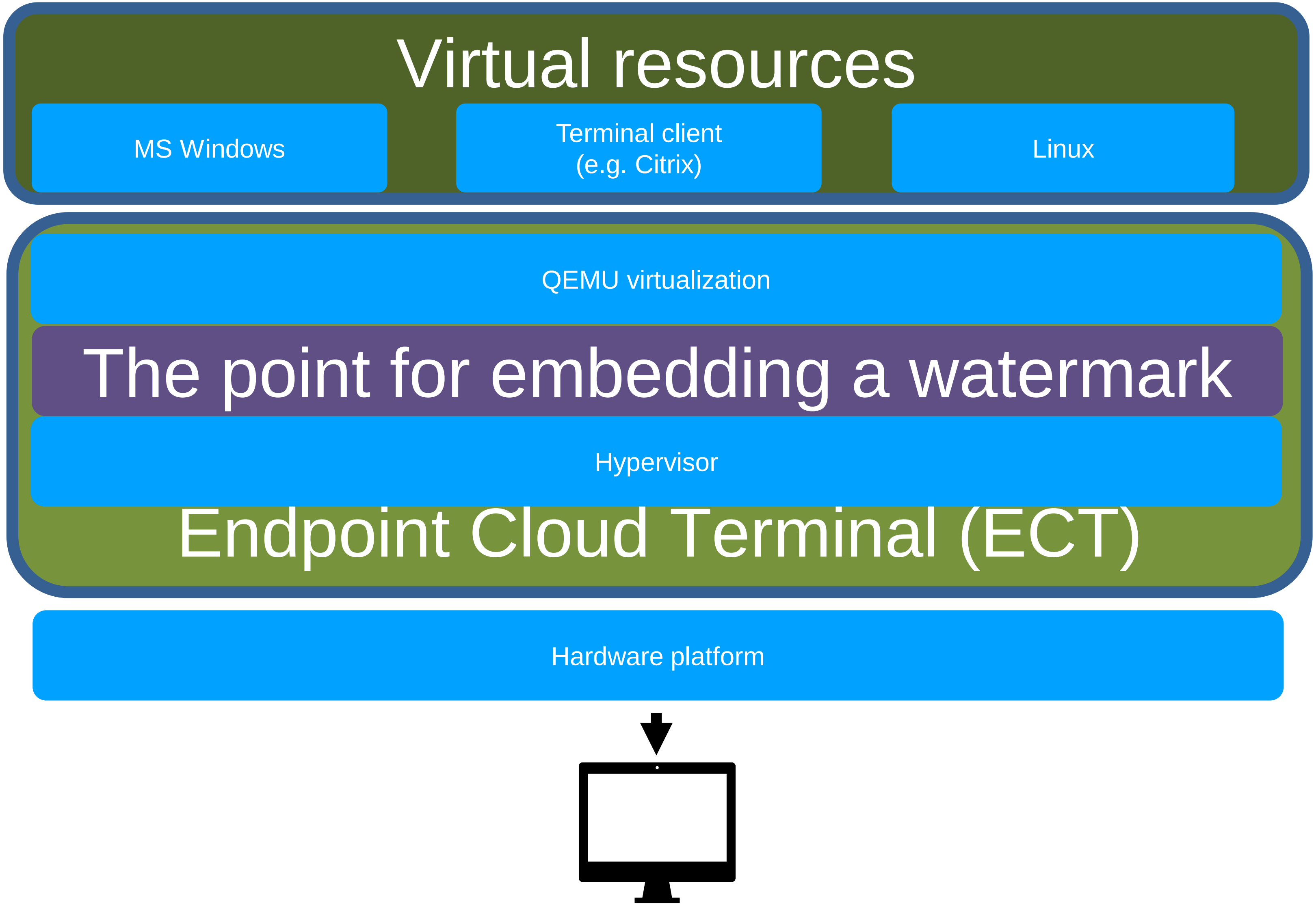
Intruder	Risks
Insider	Publishing of screen snapshots taken by a mobile devices.

Practical issues

Identifying the terminal from which the photo was taken would help not eliminate the leakage channel, but to establish the source at least

Solution and requirements

- Solution: Embedding digital watermarks into the image (steganography***)*** taken ***from an end point terminal equipped with ECT platform***
- Watermark:
  - robust
  - imperceptible
  - irreversible
- The solution:
  - Processing time ~ 41 ms (24-30 frames per second)
  - OS independent
  - Document (original) invariance
  - Minimum technical efforts to support (OS, hardware, drivers, fonts, etc.)



Virtual resources

MS Windows

Terminal client
(e.g. Citrix)

Linux

QEMU virtualization

The point for embedding a watermark

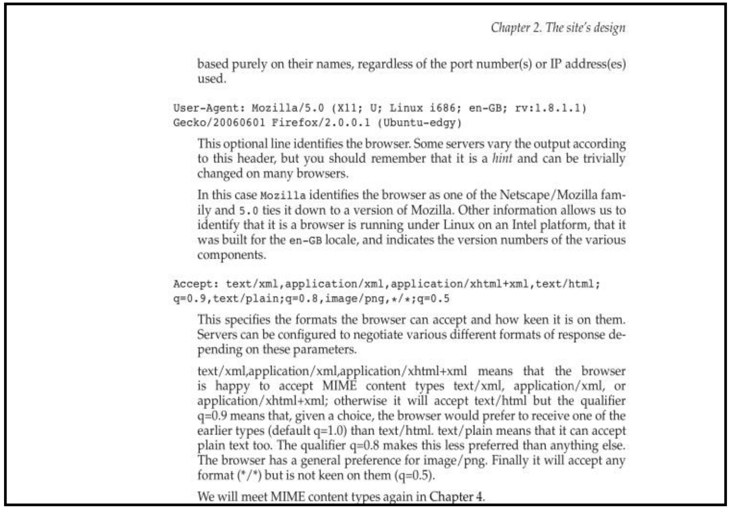
Hypervisor

Endpoint Cloud Terminal (ECT)

Hardware platform

Встраиваемый водяной знак

Timestamp	userId	secretKey	checksum	
12	10	8	2	bits



Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

```
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)
```

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case `Mozilla` identifies the browser as one of the Netscape/Mozilla family and `5.0` ties it down to a version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the `en-GB` locale, and indicates the version numbers of the various components.

```
Accept: text/xml,application/xml,application/xhtml+xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
```

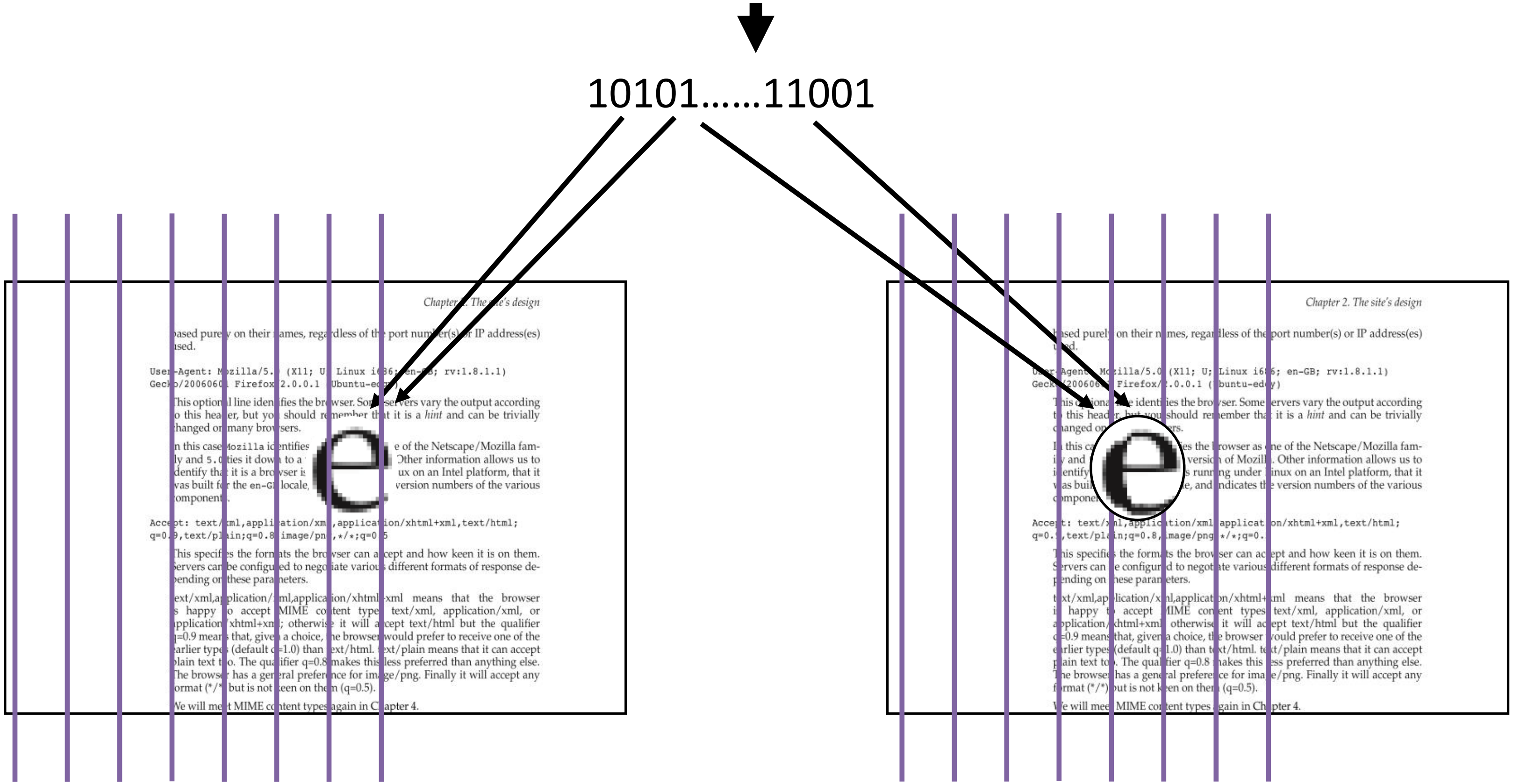
This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml+xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.

Метод встраивания ЦВЗ

Timestamp userId secretKey checksum



10101……11001

Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

```
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)
```

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case `Mozilla` identifies the browser as one of the Netscape/Mozilla family and `5.0` ties it down to a [illegible] version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the `en-GB` locale, and indicates the version numbers of the various components.

```
Accept: text/xml,application/xml,application/xhtml+xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
```

This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml+xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.

Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case Mozilla identifies the browser as one of the Netscape/Mozilla family and 5.0 ties it down to a [illegible] version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the en-GB locale, and indicates the version numbers of the various components.

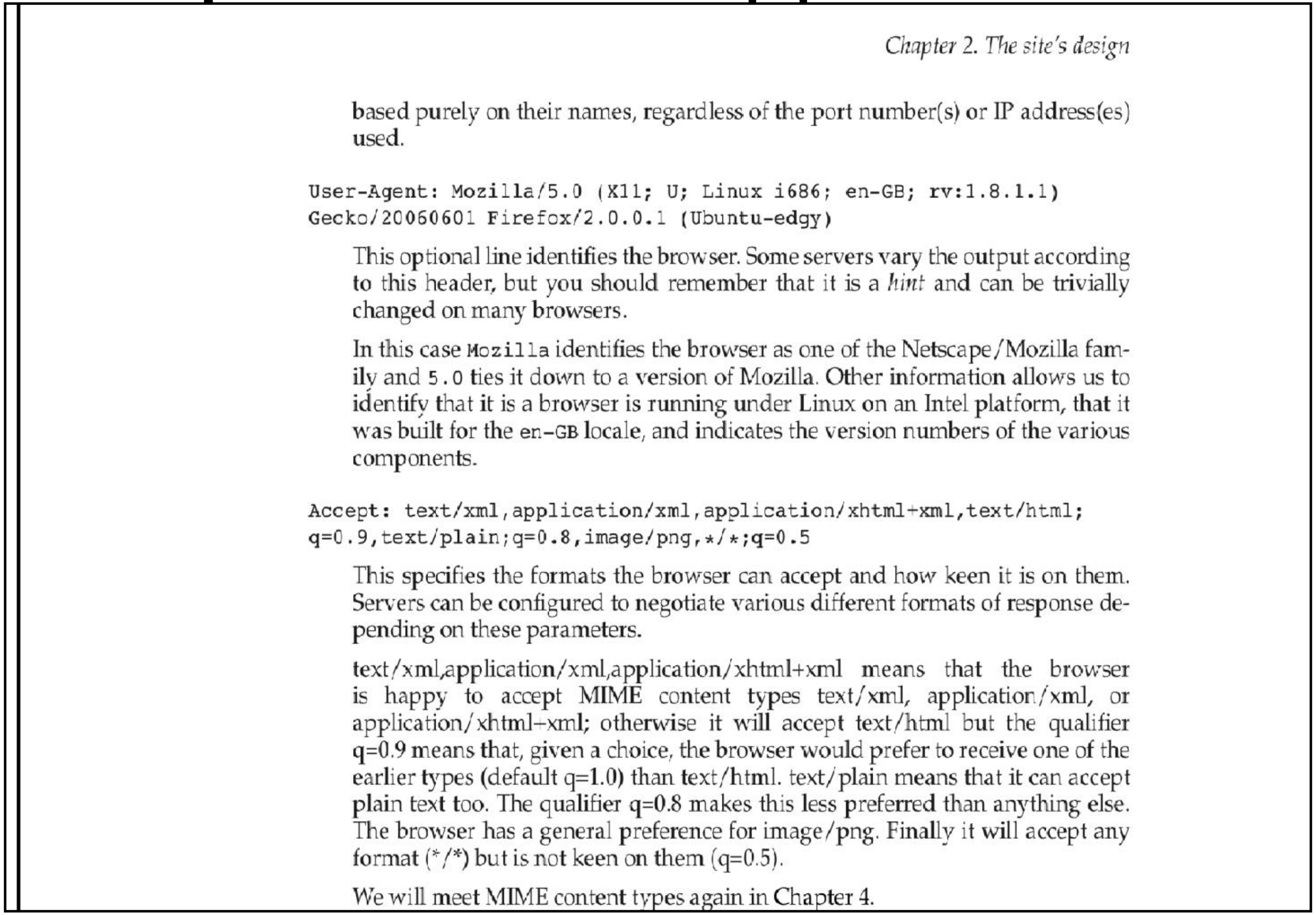
Accept: text/xml,application/xml,application/xhtml+xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5

This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml+xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.

Встроенный водяной знак



Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

```
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)
```

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case `Mozilla` identifies the browser as one of the Netscape/Mozilla family and `5.0` ties it down to a version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the `en-GB` locale, and indicates the version numbers of the various components.

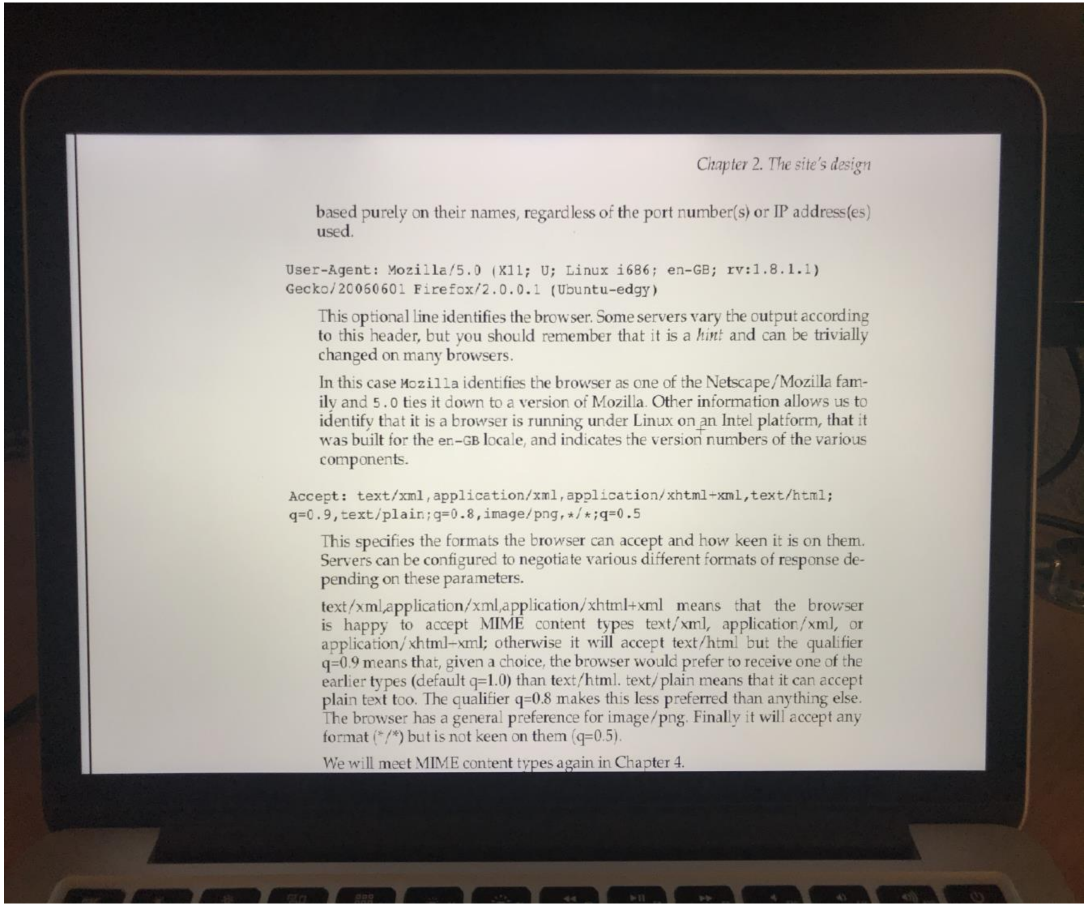
```
Accept: text/xml,application/xml,application/xhtml+xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
```

This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml+xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.

Встроенный водяной знак на снимке



Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

```
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)
```

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case `Mozilla` identifies the browser as one of the Netscape/Mozilla family and `5.0` ties it down to a version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the `en-GB` locale, and indicates the version numbers of the various components.

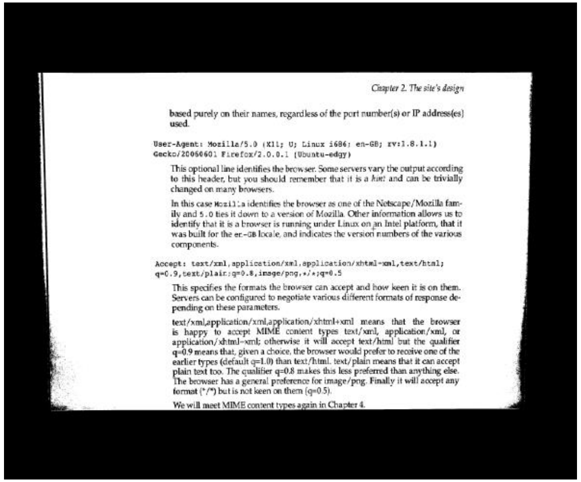
```
Accept: text/xml,application/xml,application/xhtml+xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
```

This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml+xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.

Атаки на изображение



Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

```
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)
```

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case `Mozilla` identifies the browser as one of the Netscape/Mozilla family and `5.0` ties it down to a version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the `en-GB` locale, and indicates the version numbers of the various components.

```
Accept: text/xml,application/xml,application/xhtml-xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
```

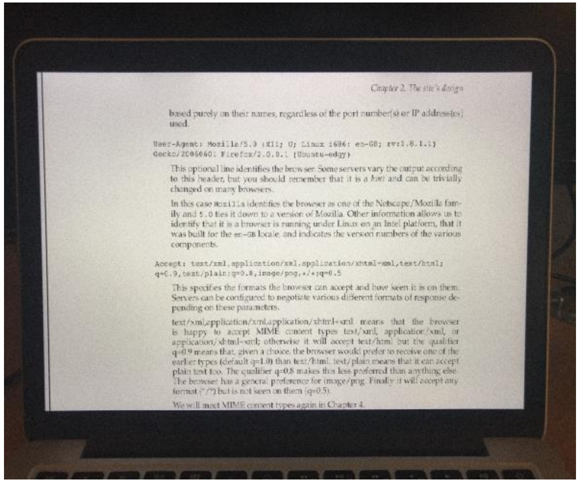
This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml-xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.

Ч&Б/Скан

Шум



Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

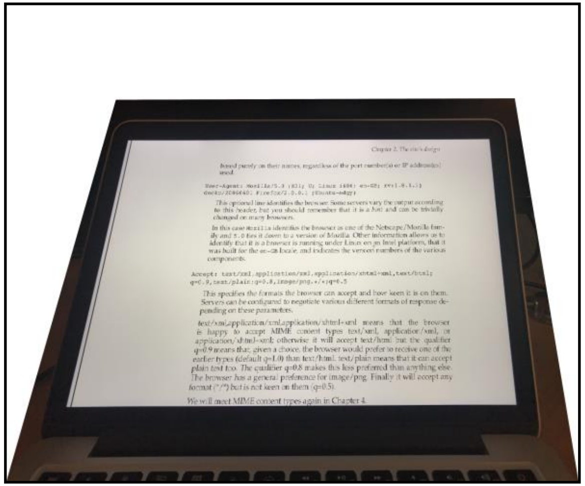
In this case Mozilla identifies the browser as one of the Netscape/Mozilla family and 5.0 ties it down to a version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the en-GB locale, and indicates the version numbers of the various components.

Accept: text/xml,application/xml,application/xhtml-xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5

This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml-xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.



Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case Mozilla identifies the browser as one of the Netscape/Mozilla family and 5.0 ties it down to a version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the en-GB locale, and indicates the version numbers of the various components.

Accept: text/xml,application/xml,application/xhtml-xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5

This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml-xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

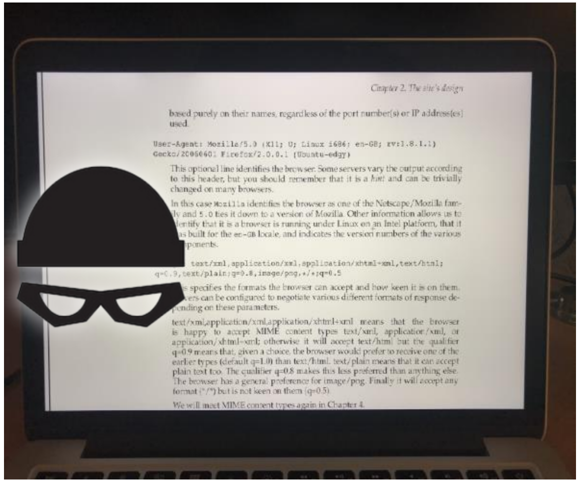
We will meet MIME content types again in Chapter 4.

Изменение

формы

Подделка

ЦВЗ



Chapter 2. The site's design

based purely on their names, regardless of the port number(s) or IP address(es) used.

User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1.1)
Gecko/20060601 Firefox/2.0.0.1 (Ubuntu-edgy)

This optional line identifies the browser. Some servers vary the output according to this header, but you should remember that it is a *hint* and can be trivially changed on many browsers.

In this case Mozilla identifies the browser as one of the Netscape/Mozilla family and 5.0 ties it down to a version of Mozilla. Other information allows us to identify that it is a browser is running under Linux on an Intel platform, that it was built for the en-GB locale, and indicates the version numbers of the various components.

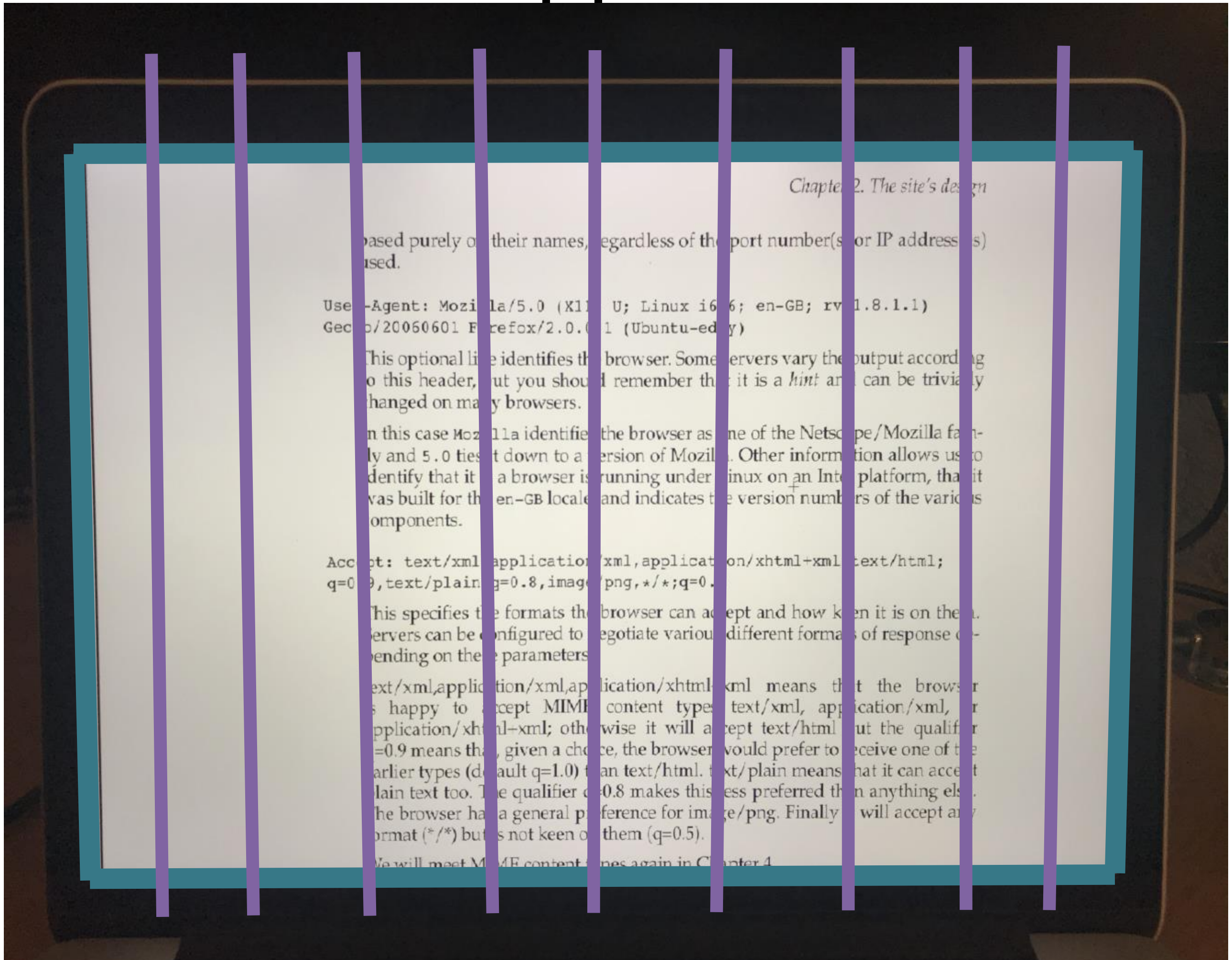
Accept: text/xml,application/xml,application/xhtml-xml,text/html;
q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5

This specifies the formats the browser can accept and how keen it is on them. Servers can be configured to negotiate various different formats of response depending on these parameters.

text/xml,application/xml,application/xhtml+xml means that the browser is happy to accept MIME content types text/xml, application/xml, or application/xhtml-xml; otherwise it will accept text/html but the qualifier q=0.9 means that, given a choice, the browser would prefer to receive one of the earlier types (default q=1.0) than text/html. text/plain means that it can accept plain text too. The qualifier q=0.8 makes this less preferred than anything else. The browser has a general preference for image/png. Finally it will accept any format (*/*) but is not keen on them (q=0.5).

We will meet MIME content types again in Chapter 4.

Чтение водяного знака



Chapte 2. The site's des gn

ased purely o their names, egardless of th port number(s or IP address s) sed.

```
Use -Agent: Mozi la/5.0 (X11  U; Linux i6 6; en-GB; rv 1.8.1.1)
Gec o/20060601 F refox/2.0.0 1 (Ubuntu-ed y)
```

his optional li e identifies th browser. Some ervers vary the utput accord ng o this header, ut you shou d remember th t it is a *hint* ar can be trivia ly hanged on ma y browsers.

n this case Moz lla identifie the browser as ne of the Netsc pe/Mozilla fa n-ly and 5.0 ties t down to a ersion of Mozil . Other inform tion allows us o dentify that it a browser is running under inux on an Int platform, tha it vas built for th en-GB locale and indicates t e version numb rs of the vario s omponents.

```
Acc pt: text/xml applicatio /xml,applicat on/xhtml+xml ext/html;
q=0 9,text/plain q=0.8,image png,*/*;q=0.
```

his specifies t e formats th browser can ac ept and how k en it is on the . ervers can be onfigured to egotiate variou different forma s of response d -pending on the e parameters

ext/xml,applic tion/xml,ap lication/xhtml xml means th t the brows r s happy to ccept MIMI content type text/xml, app ication/xml, r pplication/xht nl–xml; oth wise it will a cept text/html ut the qualif r =0.9 means tha , given a cho ce, the browser vould prefer to eceive one of t e arlier types (de ault q=1.0) t an text/html. t xt/plain means hat it can acce t lain text too. T e qualifier q 0.8 makes this ess preferred th n anything els . he browser ha a general pr ference for ima e/png. Finally will accept a y ormat (*/*) bu s not keen o them (q=0.5).

Ve will meet M ME content pes again in C apter 4

Experiments

13-inch monitor 1600x900. Density: 1 bit per 30 pixel

Text

Threshold filter

Filter	A color image	A threshold filter
Result	Yes	Yes

Altering camera angle

Angle	0°	10°	20°	30°
Result	Yes	Yes	Yes	?

Decreasing resolution

%	0 %	10 %	20 %	30 %	40 %
Result	Yes	Yes	Yes	?	No

Image with a usage of an original copy

Threshold filter

Filter	A color image	A threshold filter
Result	Yes	Yes

Altering camera angle

Angle	0°	10°	20°
Result	Yes	Yes	No

Decreasing resolution

%	0 %	10 %	20 %
Result	Yes	Yes	No

Results

- Resistance to statistical (quantization, i.e. black and white threshold filtering) and geometric (varying shooting angles and image resolution) attack methods
- Suitable to the requirements of robustness, complexity, imperceptibility, volume, irreversibility, security and verifiability
- Requirements for 24-bit message hidden in the digital watermark on a monitor screen with a resolution of 1920x1200 pixels:
  - The angles of a camera < 30 ° from the normal;
  - The resolution decrease < 30%;
  - At least 50% of a monitor screen area

Задачи распознавания

1. Сведения из теории вероятностей
2. Байесовский задачи распознавания
3. Небайесовские задачи распознавания

    3.1 Задача Неймана-Пирсона

    3.2 Задача Вальда

Десять лекций по статистическому и структурному распознаванию образов, М.И. Шлезингер, В. Главач// Kluwer Academic Publishers/ Boston/Dordrecht/London

Определения

$X, Y \ \rightarrow R$

p_{XY} – совместная вероятность события

$p_{X \ | \ Y}(x \mid y) = \frac{p_{XY}(x,y)}{\sum_{y \epsilon Y} p_{XY}(x,y)}$ - условное (апостериорное) распределение вероятности

$(p_{X \ | \ \mathrm{y}}(x) \ , y \epsilon Y)$

$(p_{x \ | \ Y}(y) \ , x \epsilon X), p_{x \ | \ Y}(y) : Y \ \rightarrow R$ - правдоподобие значения y

$p_X(x) \quad = \sum_{y \in Y} p_{XY}(x, y)$ - априорное распределение вероятностей

Определения

Байесовская задача:

X - конечное множество наблюдений , наблюдаемых параметров

K – конечное множество состояний объекта, скрытых параметров

D – множество возможных решений

$\forall\ x \in X$ и $k \in K$ задано распределение вероятностей $p_{XK} : X \times K \rightarrow R$

– совместная вероятность события

W : K× D → R – штрафная функция, или функция потерь W(k,d)

q: X→ D – стратегия решения, стратегия, или решающая функция q(x)

R(q)=E(W) – м.о. штрафа для выбранной стратегии

$R(q) = \sum_{x \in X} \quad \sum_{k \in K} p_{XK}(x, k) W(k, q(x))$

<u>Задача</u>

Для X, K, D, $p_{XK} : X \times K \rightarrow R$, *W : K× D → R*

построить байесовскую стратегию q: X→ D, минимизирующую R(q)*

Свойства

Любая байесовская стратегия делит ось вещественных чисел на $|D|$ интервалов. По наблюдению $x \in X$ принимается решение d, если отношение правдоподобия принадлежит соответствующему интервалу.

Пример: Множество D состоит из двух решений, то байесовская стратегия определяется порогом.

Вероятность ошибочного решения:

Детерминированная стратегия

Пусть объект находится в первом состоянии с вероятностью 0.9 и мы всегда называем первое состояние, то ошибка 10%, то есть когда объект находится во втором состоянии.

Рандомизированная стратегия

$0.9\ x0.1 + 0.1x0.9 = 0.18$

Свойства

Байесовская стратегия отказа от распознавания:

R(x,d) $=\sum_{k\in K} p_{K_|X}(k|x)W(k,d)$ *- частный риск*

$$W(k,d) = \begin{cases} 0, & \text{если } d = k\,, \\ 1, & \text{если } d \neq k \text{ и } d \neq \texttt{not known}\,, \\ \varepsilon, & \text{если } d = \texttt{not known}\,. \end{cases}$$

$$q(x) = \begin{cases} \operatorname*{argmin}_{d\in K} R(x,d)\,, & \text{если } \min_{d\in K} R(x,d) < R(x,\texttt{not known})\,, \\ \texttt{not known}\,, & \text{если } \min_{d\in K} R(x,d) \geq R(x,\texttt{not known})\,. \end{cases}$$

$$q(x) = \begin{cases} \operatorname*{argmax}_{k\in K} p_{K|X}(k\,|\,x)\,, & \text{если } 1 - \max_{k\in K} p_{K|X}(k\,|\,x) < \varepsilon\,, \\ \texttt{not known}\,, & \text{если } 1 - \max_{k\in K} p_{K|X}(k\,|\,x) \geq \varepsilon\,. \end{cases}$$

Небайесовские задачи распознавания

Задача Неймана-Пирсона

$\forall \varepsilon$ построить стратегию разбиения множества на два непересекающихся подмножества чтобы условная вероятность пропуска цели не превышала ε при минимальной вероятности ложной цели

Задача Вальда

Вероятность пропуска опасного состояния и вероятность ложной тревоги не могут быть одинаковыми и обычно определены заранее

Три подмножества:

X_1 , если k=1

X_2, если k=2

X_0 , если невозможно вынести решение