Feudal computing

As described by Bruce Schneier in the context of security [3], the feudal internet is the result of the move to cloud computing (Gmail) and vendor-controlled platforms (iFoo).

Feudal computing

As described by Bruce Schneier in the context of security [3], the feudal internet is the result of the move to cloud computing (Gmail) and vendor-controlled platforms (iFoo).

It is a situation where:

- we choose one or more "lords" out of a handful;
- we pledge allegiance and entrust our data to them;
- in exchange, they promise to not treat us too bad and provide some kind of service.

Feudal computing

As described by Bruce Schneier in the context of security [3], the feudal internet is the result of the move to cloud computing (Gmail) and vendor-controlled platforms (iFoo).

It is a situation where:

- we choose one or more "lords" out of a handful;
- we pledge allegiance and entrust our data to them;
- in exchange, they promise to not treat us too bad and provide some kind of service.

It is not all bad, but we have about zero power in this equation.

The surveillance morass

Technically speaking, the NSA has been conducting an all-out attack on the security of the internet, using a large budget and any means necessary.

The surveillance morass

Technically speaking, the NSA has been conducting an all-out attack on the security of the internet, using a large budget and any means necessary.

(By the way, everything aside, it is kind of interesting to know what that looks like.)

Now what?

Obviously there is no quick fix, but solutions fall into two categories.

Now what?

Obviously there is no quick fix, but solutions fall into two categories.

Legal/political: improve transparency

- regulate internet buisnesses
- rein in the NSA (ex. no secret law)

Now what?

Obviously there is no quick fix, but solutions fall into two categories.

Legal/political: improve transparency

- regulate internet buisnesses
- rein in the NSA (ex. no secret law)

Technical: "make mass surveillance more expansive" [1]

- encrypt more (opportunistic, IPsec, DNSsec)
- incorporate the new information to the threat model
- **target dispersal**

Target dispersal

> *The internet as a whole was more secure when our email was hosted by 1000 ISPs, but now there is 10. [1] 1:18:40*

Ie., the concentration and centralization of data inherent in a feudal internet creates a single point of failure, technically and socially.

Everybody, manage your own email service!

In principle, anybody can host their own email:

Everybody, manage your own email service!

In principle, anybody can host their own email:

- for free/cheap, get a DNS delegation for, say, `example.com`;

Everybody, manage your own email service!

In principle, anybody can host their own email:

- for free/cheap, get a DNS delegation for, say, `example.com`;
- rent a VPS for $5/mo, or purchase a plug server for $25;

Everybody, manage your own email service!

In principle, anybody can host their own email:

- for free/cheap, get a DNS delegation for, say, `example.com`;
- rent a VPS for $5/mo, or purchase a plug server for $25;
- install Linux and Postfix on it;

Everybody, manage your own email service!

In principle, anybody can host their own email:

- for free/cheap, get a DNS delegation for, say, `example.com`;
- rent a VPS for $5/mo, or purchase a plug server for $25;
- install Linux and Postfix on it;
- designate your server as `MX` for `example.com`.

Everybody, manage your own email service!

In principle, anybody can host their own email:

- for free/cheap, get a DNS delegation for, say, `example.com`;
- rent a VPS for $5/mo, or purchase a plug server for $25;
- install Linux and Postfix on it;
- designate your server as `MX` for `example.com`.

Et voilà! Email for `alice@example.com` is now collected by your own server, under your control.

Except it's not really possible.

Of course, that is not practical.

Except it's not really possible.

Of course, that is not practical.

Installing and maintaining a mail server is a job in itself. Few people have the skills and time required to do it.

Except it's not really possible.

Of course, that is not practical.

Installing and maintaining a mail server is a job in itself. Few people have the skills and time required to do it.

Even if you do have them, you're not going to do as good a job as Google when it comes to usability, reliability, security, ...

Except it's not really possible.

Of course, that is not practical.

Installing and maintaining a mail server is a job in itself. Few people have the skills and time required to do it.

Even if you do have them, you're not going to do as good a job as Google when it comes to usability, reliability, security, ...

In fact, the Gmail model works because it permits enormous economies of scale: millions of accounts served but only one infrastructure to maintain.

Administration as software development

However, this is not the only way in which these economies of scale can be realized.

Administration as software development

However, this is not the only way in which these economies of scale can be realized.

At its core, administration of large infrastructures (such as the one Google is using for Gmail) is a peculiar form of software development.

Administration as software development

However, this is not the only way in which these economies of scale can be realized.

At its core, administration of large infrastructures (such as the one Google is using for Gmail) is a peculiar form of software development.

There is no reason why this “software” cannot be mutualized in the same way as all of the other software we use, or why one entity should own all of the metal that runs it.

Some incarnations of this idea

The so-called "Freedom Box" [4] is a project pitched by Eben Moglen (Software Freedom Law Center) in 2010.

The idea is to bundle some privacy-enhancing free software (Tor, privoxy, ...) onto a ready-to-use plug server. Mostly vaporware at this point.

Some incarnations of this idea

The so-called "Freedom Box" [4] is a project pitched by Eben Moglen (Software Freedom Law Center) in 2010.

The idea is to bundle some privacy-enhancing free software (Tor, privoxy, ...) onto a ready-to-use plug server. Mostly vaporware at this point.

ArkOS [2] is another project with a much more limited and well-defined scope: concentrate on a few essential services such as DNS and email, and target the $25 Raspberry Pi.

Some incarnations of this idea

The so-called "Freedom Box" [4] is a project pitched by Eben Moglen (Software Freedom Law Center) in 2010.

The idea is to bundle some privacy-enhancing free software (Tor, privoxy, ...) onto a ready-to-use plug server. Mostly vaporware at this point.

ArkOS [2] is another project with a much more limited and well-defined scope: concentrate on a few essential services such as DNS and email, and target the $25 Raspberry Pi.

(Discuss viability?)

Corporate buy-in

Free Software became a viable contender when actors such as IBM started putting their weight behind it.

Corporate buy-in

Free Software became a viable contender when actors such as IBM started putting their weight behind it.

To become viable, Free "Infrastructure/software" should strive to make itself useful to this kind of corporations.

Corporate buy-in

Free Software became a viable contender when actors such as IBM started putting their weight behind it.

To become viable, Free "Infrastructure/software" should strive to make itself useful to this kind of corporations.

Fortunately, the requirement for the infrastructures of a sufficiently large organization are identical to the requirements for a global infrastructure.

Example: Kerberos, AFS, LDAP

(demo?)

Other dimensions of target dispersal

Come up with decentralized protocols for existing services (Facebook, Dropbox, . . .)


Internet Engineering Task Force.

Ietf 88 technical plenary: Hardening the internet.

http://www.youtube.com/watch?v=oV71hhEpQ20, November 2013.


Nicole Henderson.

Open source project arkos brings simplicity to self-hosting.

http://www.thewhir.com/web-hosting-news/open-source-project-arkos-brings-simplicity-to-self-hosting, November 2013.


Bruce Schneier.

When it comes to security, we're back to feudalism.

https://www.schneier.com/essay-406.html, November 2012.


Steven J. Vaughan-Nichols.

Freedom box: Freeing the internet one server at a time.

http://www.zdnet.com/blog/networking/freedom-box-freeing-the-internet-one-server-at-a-time/698, February 2011.