

CloudFront: staattisen sisällön jakaminen

the making of aws-helsinki.fi

Matti Paksula / Enemy & Sons Oy

Miksi CloudFronttiin?

- Aika halpa
- Ei ylläpitoa
- Varmasti skaalautuu

Blogi staattisena: Jekyll

- Jekyll tekee
 - staattiset html-filjet postauksista
 - atom feedin
 - hallitsee layoutin
- Kommentit JavaScriptillä ja disqussilla





AWS User Group - Helsinki

Ensimmäinen tapaaminen 22.2.2011

Kokoontumme 22.2.2011 kello 18:00 eteenpäin osoitteessa Vuorikatu 15 A, 5 krs. Sisäänpääsy [tässä kuvassa](#) näkyvästä ovesta.

Lyhyet alustukset

- Web-softan rakentaminen käyttäen palveluja EC2+SQS+ELB+RDS+S3 (Lauri Lehmijoki / Efecte Oy)
- CloudFront: staattisen sisällön jakaminen - the making of aws-helsinki.fi (Matti Paksula / Enemy & Sons Oy)
- Elastic Beanstalk, ensimmäiset hetket (Petrus Repo / Enemy & Sons Oy)
- AWS käytöönottokokemuksia (Santeri Paavolainen / Codento Oy)

Rekisteröityminen

Jotta tiedämme varautua, lähetä sähköposti osoitteeseen aws-helsinki@enemy.fi jossa kerrot tulostasi. Olisi myös toivottavaa, että saamme lisää lyhyitä (5min+) alustuksia, kerro siis mahdollisesta aiheestasi ilmoittautumisen yhteydessä.



AWS User Group - Helsinki

Ensimmäinen tapaaminen 22.2.2011

Kokoontumme 22.2.2011 kello 18:00 eteenpäin osoitteessa Vuorikatu 15 A, 5 krs. Sisäänpääsy **tässä kuvassa** näkyvästä ovesta.

Lyhyet alustukset

Instant legacy

- Web-softan rakentaminen käyttäen palveluja EC2+SQS+ELB+RDS+S3 (Lauri Lehmijoki / Efecte Oy)
- CloudFront: staattisen sisällön jakaminen - the making of aws-helsinki.fi (Matti Paksula / Enemy & Sons Oy)
- Elastic Beanstalk, ensimmäiset hetket (Petrus Repo / Enemy & Sons Oy)
- AWS käyttöönottokokemuksia (Santeri Paavolainen / Codento Oy)

Rekisteröityminen

Jotta tiedämme varautua, lähetä sähköposti osoitteeseen aws-helsinki@enemy.fi jossa kerrot tulostasi. Olisi myös toivottavaa, että saamme lisää lyhyitä (5min+) alustuksia, kerro siis mahdollisesta aiheestasi ilmoittautumisen yhteydessä.

All Things Distributed

Werner Vogels' weblog on building scalable and robust distributed systems.

New AWS feature: Run your website from Amazon S3

By Werner Vogels on February 17, 2011 3:45 PM | [Permalink](#) | [Comments \(27\)](#) | [TrackBacks \(0\)](#)

Since a few days ago this weblog serves 100% of its content directly out of the [Amazon Simple Storage Service \(S3\)](#) without the need for a web server to be involved. Because my blog is almost completely static content I wanted to run in this very simple configuration since the launch of Amazon S3. It would allow the blog to be powered by the incredible scale and reliability of Amazon S3 with a minimum of effort from my side. I know of several other customers who had asked for this greatly simplifying feature as well. I had held out implementing an alternative to my simple blog server that had been running at a traditional hosting site for many years until this preferred simple solution became available: today marks that day and I couldn't be happier about it.

The Amazon S3 team launched a new feature today that makes serving a complete (static) web site out of Amazon S3 dead simple: you set a default document for buckets and subdirectories, which is most likely an index.html document. This enables Amazon S3 to know what document to serve if one isn't explicitly requested: for example <http://www.allthingsdistributed.com> returns the index.html at the bucket level and <http://www.allthingsdistributed.com/2010/12/> the index.html from that subdirectory. The other document you can specify is a customer error page that is presented to your customers when a 4XX class error occurs (e.g. non-existent page is requested), so they get something more appropriate than just the barebones response from the browser. Click on <http://www.allthingsdistributed.com/doesnotexist.html> if you want to see what this blog's error page looks like. The background is courtesy of [@nalden](#).

All of this can be done from the AWS console as well as

A screenshot of the AWS Simple Storage Service (S3) console. The top navigation bar shows the path: Root > www.allthingsdistributed.com. Below the path are standard file operations: Copy, Move, Delete, and Search. A table lists the contents of the bucket:

Name	Size
2005	
2006	
2007	
2008	
2009	
2010	
2011	
academia	
amazoncom	
aspnet_client	
assets_c	
conferences	



Contact Info

Werner Vogels
CTO - Amazon.com

werner@allthingsdistributed.com

Other places

Follow werner on [twitter](#) if you want know what he is current reading or thinking about.

At [werner.ly](#) he posts material that doesn't belong on this blog or on twit

your website from Amazon S3

| [Permalink](#) | [Comments \(27\)](#) | [TrackBacks \(0\)](#)

your website from Amazon S3

| [Permalink](#) | [Comments \(27\)](#) | [TrackBacks \(0\)](#)

Mutkun Amazonin CDN CloudFront ois kivempi

your website from Amazon S3

| [Permalink](#) | [Comments \(27\)](#) | [TrackBacks \(0\)](#)

Mutkun Amazonin CDN CloudFront ois kivempi

Eli S3:ssa oleva sisältö CF-jakeluksi

your website from Amazon S3

| [Permalink](#) | [Comments \(27\)](#) | [TrackBacks \(0\)](#)

Mutkun Amazonin CDN CloudFront ois kivempi

Eli S3:ssa oleva sisältö CF-jakeluksi

Kysytään tätä Werneriltä



Werner Vogels said:

Mr. C. you are correct, I could do that and it would work fine. However a lot of my content is long tail, and isn't accessed very frequently and as such just blindly serving it through cloudfont kind-off violates my frugal engineer's mind :-). I am planning to reorganize into long tail and popular, such the latter can be served through cloudfont always.

But you are right in that I could just enable cloudfont always and just not worry about it :-)

s3cmd

s3cmd

```
s3cmd sync . s3://aws-helsinki.fi \
```

s3cmd

```
s3cmd sync . s3://aws-helsinki.fi \  
--delete-removed \  
"
```

s3cmd

```
s3cmd sync . s3://aws-helsinki.fi \  
--delete-removed \  
--reduced-redundancy \  
"
```

s3cmd

```
s3cmd sync . s3://aws-helsinki.fi \  
--delete-removed \  
--reduced-redundancy \  
--exclude=".git/*"
```

s3cmd

```
s3cmd sync . s3://aws-helsinki.fi \  
  --delete-removed \  
  --reduced-redundancy \  
  --exclude=".git/*"
```

<http://aws-helsinki.fi.s3.amazonaws.com/index.html>

S3 REST API -> Website

The screenshot shows a user interface for configuring an S3 bucket as a website. At the top, there are four tabs: 'Permissions', 'Website' (which is highlighted with a yellow background), 'Logging', and 'Notifications'. Below the tabs, there is a message: 'You can host your static websites entirely out of Amazon S3. Once your bucket has been configured as a website, you can access it via the Amazon S3 website endpoint for your bucket.' Underneath this message, several settings are listed: 'Enabled:' followed by a checked checkbox; 'Index Document:' followed by a text input field containing 'index.html'; 'Error Document:' followed by an empty text input field; and 'Endpoint:' followed by a blue hyperlink: 'http://aws-helsinki.fi.s3-website-us-east-1.amazonaws.com/'.

Enabled:

Index Document: index.html

Error Document:

Endpoint: <http://aws-helsinki.fi.s3-website-us-east-1.amazonaws.com/>

Sallitaan lukeminen

Bucket Policy Editor Cancel

Policy for Bucket : "aws-helsinki.fi"

Add a new policy or edit an existing bucket policy in the text area below.

```
{  
    "Version": "2008-10-17",  
    "Id": "ba127104-4646-4359-aac5-7433f989a46f",  
    "Statement": [  
        {  
            "Sid": "AddPerm",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "*"  
            },  
            "Action": "s3:GetObject",  
            "Resource": "arn:aws:s3:::aws-helsinki.fi/*"  
        }  
    ]  
}
```

[AWS Policy Generator](#) | [Sample Bucket Policies](#) [Save](#) [Delete](#) [Close](#)

Luodaan CF jakelu

Luodaan CF jakelu

```
s3cmd cfcreate s3://aws-helsinki.fi \
```

Luodaan CF jakelu

```
s3cmd cfcreate s3://aws-helsinki.fi \
--cf-default-root-object=index.html \
```

Luodaan CF jakelu

```
s3cmd cfcreate s3://aws-helsinki.fi \
--cf-default-root-object=index.html \
--cf-add-cname=www.aws-helsinki.fi
```

Luodaan CF jakelu

```
s3cmd cfcreate s3://aws-helsinki.fi \
--cf-default-root-object=index.html \
--cf-add-cname=www.aws-helsinki.fi
```

Origin:	s3://aws-helsinki.fi/
DistId:	cf://E2V6PA0G4TXUR8
DomainName:	d3chn75r1q03tp.cloudfront.net
Status:	Deployed
Enabled:	True

Luodaan CF jakelu

```
s3cmd cfcreate s3://aws-helsinki.fi \
--cf-default-root-object=index.html \
--cf-add-cname=www.aws-helsinki.fi
```

Origin:	s3://aws-helsinki.fi/
DistId:	cf://E2V6PAOG4TXUR8
DomainName:	d3chn75r1q03tp.cloudfront.net
Status:	Deployed
Enabled:	True

```
s3cmd cfmofify cf://E2V6PAOG4TXUR8
--cf-add-cname=aws-helsinki.fi
```

DNS

Name	Data	Type
aws-helsinki.fi.	d3chn75r1q03tp.cloudfront.net.	CNAME
www	aws-helsinki.fi.	CNAME
aws-helsinki.fi.	ns1.slicehost.net.	NS
aws-helsinki.fi.	ns2.slicehost.net.	NS
aws-helsinki.fi.	ns3.slicehost.net.	NS

DNS

Name	Data	Type
aws-helsinki.fi.	d3chn75r1q03tp.cloudfront.net.	CNAME
www	aws-helsinki.fi.	CNAME
aws-helsinki.fi.	ns1.slicehost.net.	NS
aws-helsinki.fi.	ns2.slicehost.net.	NS
aws-helsinki.fi.	ns3.slicehost.net.	NS

Domainilla ei A-recordia, eli jotkin
mailiserverit ei pidä

DNS

Name	Data	Type
aws-helsinki.fi.	d3chn75r1q03tp.cloudfront.net.	CNAME
www	aws-helsinki.fi.	CNAME
aws-helsinki.fi.	ns1.slicehost.net.	NS
aws-helsinki.fi.	ns2.slicehost.net.	NS
aws-helsinki.fi.	ns3.slicehost.net.	NS

Domainilla ei A-recordia, eli jotkin
mailiserverit ei pidä

Eikä myöskään RFC-ihmiset?

DOMAIN NAMES - CONCEPTS AND FACILITIES

1. STATUS OF THIS MEMO

This RFC is an introduction to the Domain Name System (DNS), and omits many details which can be found in a companion RFC, "Domain Names - Implementation and Specification" [RFC-1035]. That RFC assumes that the reader is familiar with the concepts discussed in this memo.

A subset of DNS functions and data types constitute an official protocol. The official protocol includes standard queries and their responses and most of the Internet class data formats (e.g., host addresses).

However, the domain system is intentionally extensible. Researchers are continuously proposing, implementing and experimenting with new data types, query types, classes, functions, etc. Thus while the components of the official protocol are expected to stay essentially unchanged and operate as a production service, experimental behavior should always be expected in extensions beyond the official protocol. Experimental or obsolete features are clearly marked in these RFCs, and such information should be used with caution.

The reader is especially cautioned not to depend on the values which appear in examples to be current or complete, since their purpose is primarily pedagogical. Distribution of this memo is unlimited.

2. INTRODUCTION

This RFC introduces domain style names, their use for Internet mail and host address support, and the protocols and servers used to implement

November 1987

November 1987



**THIS IS WHY
WE CAN'T HAVE
NICE THINGS**

Luigi Montanez said:

Redirecting all my users to www.{mydomain}.com is a dealbreaker. Is there any technical workaround to serve from S3 using {mydomain}.com sans www?

Roland Turner said:

...the currently made-up CloudFrontDistribution tag has been implemented) with something like:

```
<ResourceRecordSet>
  <Name>example.com</Name>
  <Type>A</Type>
  <ResourceRecords>
    <ResourceRecord>
<CloudFrontDistribution>tho9reeMTap3quae.cloudfront.net</CloudFrontDistribution>
    </ResourceRecord>
  </ResourceRecords>
</ResourceRecordSet>
```

Petrus Repo said:

Jos CNAME kerran toimii, niin onko sillä väliä?

Gotchas

- CloudFrontin päivitysnopeus “24h”
 - DNS:n TTL pieneksi ja uusi jakelu..?
- Googlen DNS ja CDN?

Mark said:

You can implement that commenting functionality, too:

Using Amazon's SQS, create a public-writeable queue and have all the comments posted to it.

```
"Effect": "Allow", "Principal": {"AWS": "*"}, "Action":  
"sns:Publish",
```

If you want to be notified about new posts, you could do that by SNS.

Finally you fetch comments from the queue and recreate your pages into S3 again. Just like in any moderated blog.

thx