

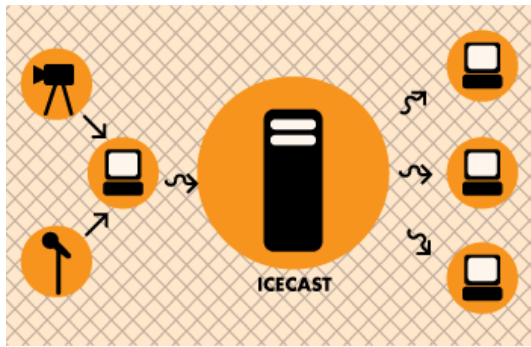


FLOSS
MANUALS

Table of Contents

Icecast.....	1
Wat is streaming?.....	3
Verschillende vormen van Streaming.....	3
Download and play / Download en speel af.....	3
Progressive download/Progressieve download.....	3
True streaming / Echte streaming.....	3
Static file streaming / Statisch bestand streamen.....	4
Live streaming.....	4
Encoding /coderen.....	4
Compression.....	4
Lossy compression.....	5
Codecs.....	5
Bitrate.....	5
Delivery / levering.....	5
Streaming servers.....	5
Icecast installeren van de bron (Linux).....	7
De opgeslagen bestanden uitpakken.....	8
Paths/paden en Logfiles/logbestanden.....	10
Icecast opstarten.....	11
Configuring Icecast.....	12
Comments.....	12
Changing the Config File.....	13
Authentication.....	13
Hostname and Port.....	13
Accounts Aanmaken.....	14
Standaard Account Instellingen.....	15
wachtwo0rd.....	15
max-listeners (maximaal aantal luisteraars).....	15
dump-file (dump bestand)	15
intro.....	15
Credits.....	16
Licence.....	16
Authors.....	16

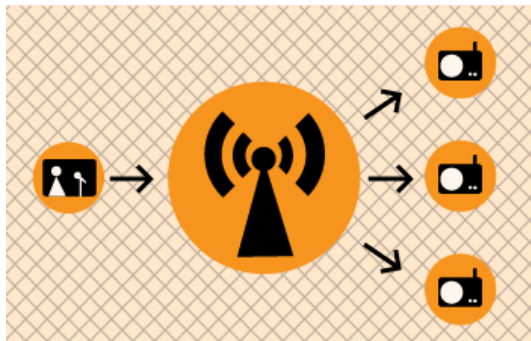
Icecast



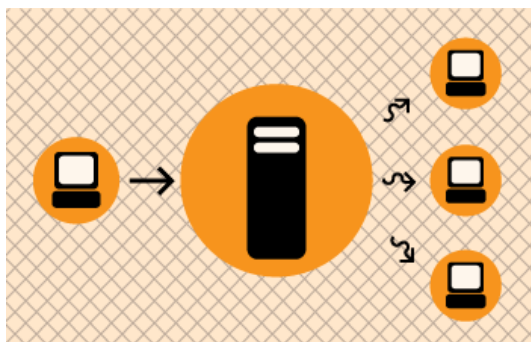
In de wereld van online audi en video is er een groep technologieen en technieken die streaming genoemd worden. Streaming is kortweg het afleveren van audio en video over het internet in realtime (de ontvangst is tegelijkertijd met de uitzending). Traditioneel wordt streaming gebruikt voor het aanbieden van live internet radio, of van het uitzenden van evenementen over het internet zodat je ze thuis kunt bekijken.

Als er over streaming gepraat wordt, worden vaak termen gebruikt uit de broadcast industrie, omdat live internet audio en video vaak gezien worden als vergelijkbaar met tv en radio. Zo wordt de term 'internet audio' gebruikt voor live online audio. Deze termen kunnen bruikbaar zijn om te begrijpen wat het doel van streaming is. maar we hoeven ze niet te letterlijk te nemen. Het is wel handig om naar deze modellen te verwijzen als we de rol van Icecast willen uitleggen.

Met broadcast/uitzend tv of radio, is er een zender die een signaal stuurt van de studio, naar je tv of radio.



Met internet-broadcasting (streaming) vervang je de zender met een **streaming server**.



Icecast is zo'n software. Je kunt er je live audio en video mee over het internet sturen in 'realtime'. Dit type software wordt vaak een Streaming Server genoemd. Je hebt daarnaast ook een stream encoder nodig, die de originele audi of video stream naar Icecast kan sturen. Icecast distribueert de stream, het maakt de stream zelf

niet - dat is de taak van de encoder.

Icecast is een erg robuuste technologie. Het ondersteunt de levering van verschillende formaten van streaming audio (waaronder MP3 en Ogg Vorbis) en het ondersteunt de verzending van Ogg Theora voor streaming video. Icecast werkt op Linux en Windows. Als je het op Linux wilt installeren zal je wat ervaring nodig hebben. Je moet dan vooral weten hoe je software installeert van de command line, en hoe je text-gebaseerde configuratiebestanden aanpast. Het installeren op Windows is wat makkelijker, maar je moet nogsteeds tekst-gebaseerde configuratiebestanden aan kunnen passen.

Het installeren en draaien van Icecast is niet aanbevolen voor beginnende gebruikers.

Wat is streaming?

Streaming is de term die gebruikt wordt voor de live-ontvangst van audio en video over het internet. Met Streaming Media kan je live audio of video over het internet sturen, waardoor het internet ook een broadcast medium wordt. Content kan ook in streaming bestandsformaten gearhiveerd worden, waardoor internetgebruikers een evenement achteraf nog kunnen bekijken.

Verschillende vormen van Streaming

Er zijn drie manieren waarop audio en video over het internet ontvangen kunnen worden: Download and play, Progressieve Download, en True Streaming.

Download and play / Download en speel af

Om "Download and Play"-media te bekijken/luisteren moet je eerst het hele bestand op je computer opslaan voordat je het kunt afspelen. D&P kan daarom niet gebruikt worden voor live uitzendingen, maar het is wel een goede manier om media (audio en video) van een hoge kwaliteit te versturen over elke bandbreedte. Een goede kwaliteit film, bijvoorbeeld, kan gedownload worden met elke internet verbinding. Maar, omdat je de film niet kunt afspelen totdat je het hele bestand gedownload hebt, wordt de download-tijd een belangrijke factor. Een DVD film te downloaden (gewoonlijk zo'n 4.6 GB) met een opbel-verbinding (56kbps) zou dan meer dan 8 dagen duren. De meeste mensen zouden dit erg vervelend vinden!

D&P-media is *geen* streaming media (hier later meer over), maar het is een stuk populairder dan streaming. Bestandsuitwisselings-programma's & netwerken als Napster en Limewire zijn D&P-mechanismen. Omdat meer en meer mensen toegang hebben tot meer bandbreedte, wordt het steeds populairder om films en muziek uit te wisselen via deze netwerken.

Progressive download/Progressieve download

Progressieve download geeft je de mogelijkheid om de media af te spelen terwijl je het aan het downloaden bent. Het is handig, maar minder efficiënt dan True Streaming, en mist sommige functionaliteiten zoals 'multiple bitrate encoding' (het coderen voor verschillende bitrates). Het voordeel van het gebruiken van deze vorm i.p.v. Download & Play is dat je hiermee niet hoeft te wachten tot je hele bestand gedownload is voordat je het kunt spelen. Er zijn wel wat beperkingen, bijvoorbeeld, als het bestand niet zo snel downloadt als dat je hem afspeelt, zal het afspelen stoppen totdat er weer genoeg gedownload is. Dit kan irritant zijn en is vaak een goede reden om voor True Streaming te kiezen.

Quicktime (Apple's Streaming programma), noemt Progressieve Download "Fast Start". Een van de problemen van het leren van streamen, is het onderscheiden van de verschillende methodes, omdat iedere software-aanbieder hun eigen jargon aanmaakt. Maar tijdens deze handleiding zal ik de meest standaard termen gebruiken waar mogelijk.

True streaming / Echte streaming

Dit geeft de gebruiker de mogelijkheid om de media te beluisteren of bekijken terwijl het live over het internet ontvangen wordt. True Streaming is de enige manier van Streamen die je de mogelijkheid geeft om live uit te zenden. Hoewel er verschillende termen gebruikt worden om True Streaming te beschrijven (zoals webcasting, livecasting, web tv of net.radio) is een veel gebruikte term: "streaming media" of nog vaker gewoon "streaming".

Het verschil tussen PD en TS is niet altijd duidelijk. Voor gebruikers met een snelle internetverbinding kan het lijken dat PD en TS hetzelfde zijn.

Binnen D&P, PD en TS zijn er twee duidelijke varianten: static file streaming (statisch bestand streamen) en live streaming (live streamen).

Static file streaming / Statisch bestand streamen

Het live ontvangen van vooraf-opgenomen bestanden over het internet. Als we het over online-media archieven hebben, hebben we het vaak over statisch bestand streamen. Een archief van videokunst die opgeslagen zijn als streaming bestanden, is een voorbeeld van een collectie van statisch-bestand-streamen-bestanden. Deze content is ook bekend als 'on demand' of 'gearchiveerde' content. Het kan aangeboden worden via D&P, PD of TS.

Live streaming

Het afleveren van live audio of video over het internet. De gebruiker kan het evenement tegelijk meeleven. Er zijn hier veel voorbeelden van, zoals online radio of het kijken van live optredens. Je kunt alleen live streamen met True Streaming.

Deze handleiding zal zich vooral bezighouden met True Streaming van zowel live als gearchiveerde content.

Encoding /coderen

Om statische bestanden over het internet te streamen, moeten de bestanden gecomprimeerd en ge-codeerd worden naar een 'streaming bestandsformaat'. Dit is vergelijkbaar met het omzetten van CD's naar MP3's. In dit proces wordt CDgeluid omgezet door de codeer-software van een CD audio formaat naar een 'streaming audio formaat' (MP3). Wat er dan o.a. gebeurt is dat de gegevens/data in het bestand gecomprimeerd worden door de kwaliteit en bestandsgrootte te verkleinen, en de data wordt omgezet in een 'streaming bestandsformaat' (b.v. MP3).

Compression

Er zijn twee vormen van comprimeren - lossy en lossless (zonder loss).

Als het codeer proces het bronbestand comprimeerd zodat het live over een internet-verbinding gestuurd kan worden, vermindert dit proces de kwaliteit van de audio en video. Hoe meer gecomprimeerd het bestand is, hoe lager de bandbreedte is die nodig is om het bestand te spelen, maar hoe lager ook de kwaliteit is. Je moet een compromis zoeken, waar het niveau van comprimeren zowel acceptabele audio en video oplevert, en ook nog te versturen is over het internet.

Het live-coderen is vergelijkbaar, behalve dat audio en/of video input gecodeerd wordt in plaats van een bestand. Met die proces stuurt de codeer-software de gecodeerde data in een continue stream naar de server.

Lossless compression

Dit is het proces waarmee data gecomprimeerd wordt tot een kleiner bestand zonder data te verwijderen. Stel je een plastic zak voor met een object erin. Als je de lucht uit de zak haalt door een vacuum te maken, wordt het object in de zak niet aangetast, terwijl de afmeting van de zak wel kleiner wordt.

Lossy compression

Dit wordt soms ook wel "Perceptual Encoding" (Waarnemings encoderen) genoemd. Dit is het proces waarbij data weggegooid wordt om de bestandsgrootte kleiner te maken. De compressie-algoritmes die gebruikt worden zijn erg ingewikkeld en proberen ervoor te zorgen dat het bestand er voor jou nog hetzelfde uitzien en klinkt, maar dat er toch zoveel mogelijk van de bestandsgrootte wordt afgeschaafd.

Codecs

De algoritmes die gebruikt worden om audio en video te comprimeren en coderen om het uiteindelijke bestandsformaat te creëren staan bekend als 'codecs'. Het woord is een samenstelling van de termen: compress (comprimeer) en decompress (ont-comprimeer). De codeer-software gebruikt een codec om de streaming content te 'comprimeren' voor het afleveren over het internet, en de software van de gebruiker gebruikt een codec om de content te 'ontcomprimeren' om het af te kunnen spelen.

De codec is wiskundige tovenaarschap en is de essentie van het streamen. Elke codec werkt op zijn eigen manier, gebaseerd op de aannames van de ontwikkelaar, en ze hebben ook zo allemaal hun eigen licenties.

Bitrate

Compressie is het proces van het verkleinen van de hoeveelheid data.

Met Multiple bitrate encoding kunnen de server en de speler samen de beste kwaliteit (hoogste bandbreedte) stream kiezen die geleverd wordt. De speler krijgt dan de beste kwaliteit stream die hij kan ontvangen over zijn eigen internet-verbinding. Multiple Bitrate Encoding produceert maar 1 ge-codeerde stream.

Het kiezen van de bitrate zal afhangen van verschillende factoren waaronder: De verbinding van de doelgroep, Voor video: De afmetingen (in pixels). De hoeveelheid beweging in de video. Het contrastniveau. De kwaliteit van de camera en de lens. En voor Audio en Video: De soort audio die wordt gecodeerd (zoals stem, stereo, muziek).

Delivery / levering.

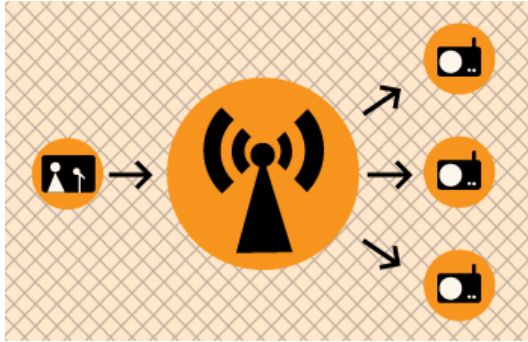
Streamende statische/gearchiveerde bestanden kunnen via een gewone webserver aangeboden worden. Dit is vaak de goedkoopste manier om content aan te bieden op een kleine schaal. Deze methode kan niet voor live streaming gebruikt worden, en biedt niet de mogelijkheden van True Streaming zoals multiple bitrate encoding. Met deze methode krijg je ook vaker 'time-outs' en je kunt er niet zoveel mensen tegelijkertijd mee bedienen als met True Streaming.

Om live streaming mogelijk te maken en voor optimale functionaliteiten en efficiëntie is een streaming media server nodig. Deze server gebruikt de standaard server hardware, maar met de benodigde streaming server softwares. Je kunt ook een streaming server installeren op dezelfde machine als een bestaande webserver.

Streaming servers

Een goed voorbeeld om te snappen wat streaming is, is om te denken aan een radiostation. Een radiostation bestaat uit drie onderdelen: - een *studio*, een *zender* en de *ontvangers* die je

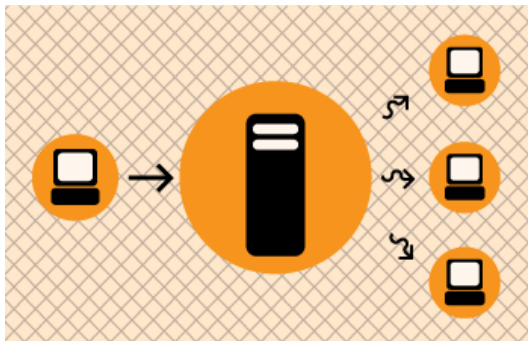
publiek heeft.



Hierboven zie je een eenvoudig diagram dat toont hoe een zendend radio station werkt. Het radiostation is de bron van de audio. In deze ruimte zijn gewoonlijk cd-spelers, minidisk spelers, platenspelers en mengpanelen. Dan wordt uit de studio een audiosignaal naar de zender gestuurd. Dit kan gedaan worden via een kabel (soms een 'landline' genoemd) of via een microgolf verbinding. De zender zendt de audio dan via FM, zodat radio-ontvangers het signaal op kunnen vangen en spelen.

Radio werkt zo omdat ze proberen hun studio-geluid naar zoveel mogelijk mensen te sturen. Als je je een radiostudio zonder zender voorstelt, zouden ze een behoorlijk klein publiek hebben, alleen de mensen in de studio zouden het dan kunnen horen. De zender werkt daardoor als distributeur, zodat meer mensen via hun 'radio's' kunnen verbinden, en het potentiële publiek vergroot wordt.

Streaming wordt ook gebruikt om je audio voor meer mensen beschikbaar te stellen.



De computer vervangt dan de radiostudio, de streaming server vervangt de zender, en je luisteraars luisteren via hun computer naar de server in plaats van met radio-ontvangers naar het FM-kanaal.

De vergelijking gaat nog verder: Een grotere radio-zender en meer bandbreedte op je streaming server zorgen allebei voor meer potentiële luisteraars.

Icecast installeren van de bron (Linux)

Dit onderdeel bespreekt de installatie van Icecast op een Linux machine, door gebruik te maken van de bronnen. Dit betekent dat je het programma moet compileren door gebruik te maken van de commandline.

Op de beginpagina van de Icecast website staat een "Download" link, als je hierop klikt, kom je op het Downloadonderdeel van de site:

<http://www.icecast.org/download.php>.

Op het moment van schrijven is de laatste versie 2.3.1. maar het is waarschijnlijk dat je met een andere versie werkt. Dit maakt niet zoveel uit, het proces zal grotendeels hetzelfde zijn. Je zult zien dat er verschillende versies van de software beschikbaar zijn, er zijn bijvoorbeeld een paar versies voor Redhat, en er is een Windows versie. Het bestand dat we nu willen is het 'source' of 'bron' bestand. Dit is een gecomprimeerd bestand, met alle installatie bestanden erin.

Download de 'source' bestanden, deze kan je waarschijnlijk vinden onder 'all/alle' van het 'Platforms/platformen' onderdeel van de Icecast site. Je kunt het ook herkennen aan de .tar.gz uitgang. In mijn voorbeeld download ik het bestand van <http://www.icecast.org>

```
icecast-2.3.1.tar.gz
```

Ok.. Ik hoop dat je dit bestand ergens hebt opgeslagen waar je het kunt terugvinden. Ik sla mijn installatiebestanden vaak op in de home-map in een map met de naam src. Je kunt je bestanden overal opslaan waar je wilt, maar het is het handigst om niet je hele bestandssysteem te vervuilen met installatiebestanden overal. Ik raad je dan ook aan om het in je home-map te gooien. De home-map is overigens de map met je gebruikersnaam erop, die in de /home directory zit. Om in deze home directory to komen kan je het volgende in je terminal venster typen:

```
cd /home/username
```

Waar 'username' *jouw gebruikersnaam* is, bijvoorbeeld: mijn gebruikersnaam is 'adam' dus ik type:

```
cd /home/adam
```

of probeer dit:

```
cd ~
```

Het **cd** commando betekent verander directory, en je gebruikt dit commando vaker als je door je bestandssysteem heen navigeert. Als je in je home map bent, zoek dan naar een map met de naam src. Type hiervoor:

```
ls -al
```

Dit commando: **ls** betekent 'list/lijs', as je het typt worden al je bestanden in je home map getoond. Als je je src map nog niet ziet, type dan het volgende:

```
mkdir src
```

Het **mkdir** commando betekent **maak directory**, als je het bovenstaande typt, wordt er een map aangemaakt met de naam 'src'. Nu moet je van map veranderen (cd) naar de nieuwe map. Daar moet je de bronnen plaatsen als je gene beter plek hebt. Als je de bronnen nog niet opgeslagen hebt, sla ze dan hierop. Als je ze al hebt kan je de bronnen naar deze map verplaatsen door het **mv** commando te gebruiken.

```
mv /home/username/icecast-2.0.2.tar.gz /home/username/src
```

In het bovenstaande voorbeeld heb ik per ongeluk de bronnen in de /home/adam map gezet, en ik verplaats ze nu naar mijn nieuwe src map.

De opgeslagen bestanden uitpakken.

Nu zullen we het net-opgeslagen-bestand uitpakken. Type hiervoor het volgende: (er vanuitgaand dat je de 2.3.1. versie van Icecast heb, vervang anders de bestandsnaam hieronder voor de jouwe):

```
tar -zxvf icecast-2.0.2.tar.gz
```

Je zult nu een heleboel tekst in je terminal venster zien, dit is een lijst van alle bestanden die net uitgepakt zijn. Type als je meer wilt leren over het tar commando het volgende:

```
man command.
```

Als je nu het ls - al commando typt, zie je hopelijk een nieuwe map:

Je moet nu met cd naar den ieuw mpa gaan (in mijn geval heet deze map icecast-2.3.1).

```
cd icecast-2.3.1
```

Type dan het volgende:

```
./configure
```

Dit is het standaard configuratie commando, dat je zo'n beetje altijd gebruikt als je iets van bron installeert op Linux. Als je gelukt hebt zie je een hoop tekst door de terminal heen scrollen.. dit is het configuratie script dat kijkt of alles op je computer klopt voordat het de software installeert. Als het goed gaat heb je een fijn snel configuratie proces zonder problemen.

Type dan het volgende:

```
make
```

Het make commando compleert een binary van de bronnen. Dit betekent dat het configuratiescript heeft onderzocht hoe je de software voor je systeem maakt, en het make commando gebruikt deze informatie om de software te compileren/maken. Als het maakproces goed gaat, krijg je een mooie lege terminal zonder foutmeldingen.

ok! dus.. als je problemen had, en je dit niet gezien hebt, heb je hulp nodig. Ik kan hier niet meer tips geven, behalve dat je via een zoekmachine naar antwoorden op zoek kan gaan. Ik kopieer altijd de foutmelding die ik krijg, en plak deze in mijn favoriete zoekmachine, en hoop op goede resultaten. Hopelijk was er iemand anders met hetzelfde probleem die de oplossing online heeft gezet, of staat het in een nieuwsgroep. In ieder geval, zijn de problemen die kunnen ontstaan nogal breed, en is het niet mogelijk om ze allemaal te voorspellen.

Als het goed gaat is er nog 1 commando dat je in kunt tikken:

```
make install
```

Dit commando installeert de gecompileerde software op de goede plaats op je systeem. Om te controleren of dit allemaal goed ging kan je het volgende typen:

```
icecast
```

Dit is het commando dat je vers geïnstalleerde Icecast 2 server opstart...

Paths/paden en Logfiles/logbestanden

Soms vind ik dat Icecast foutmelding geeft als het de logbestanden probeert te maken. De foutmelding ziet er dan zo uit:

```
FATAL: could not open error logging
FATAL: could not open access logging
FATAL: Could not start logging
```

Als dit gebeurt open ik mijn favoriete tekst-bewerkings-programma en bekijk het 'icecast.xml' file. Daar zoek ik deze regels:

```
<logdir>/usr/local/var/log/icecast</logdir>
  <webroot>/usr/local/share/icecast/web</webroot>
  <adminroot>/usr/local/share/icecast/admin</adminroot>
```

Als ik deze waarden verander zodat ze naar mijn home map verwijzen:

```
<logdir>/home/adam</logdir>
  <webroot>home/adam</webroot>
  <adminroot>/home/adam</adminroot>
```

Je gebruikt hier natuurlijk de naam van je eigen map. Verander ook de waarden in het volgende gedeelte van het configuratiebestand zodat je het volgende leest:

```
<logging>
  <accesslog>access.log</accesslog>
  <errorlog>error.log</errorlog>
  <loglevel>4</loglevel> <!-- 4 Debug, 3 Info, 2 Warn, 1 Error -->
</logging>
```

En in je home map type je het volgende:

```
touch access.log error.log
```

Het touch commando maakt wat lege bestanden waar icecast2 heen kan schrijven. Hierna doet Icecast het vaak gewoon goed. Als je een text-verwerker pakt en dit wachtwoord veranderd etc, en Icecast draaide al, moet je Icecast stopzetten (ctrl en c tegelijkertijd intikken) en daarna Icecast herstarten.

Icecast opstarten

Icecast wordt helemaal aangestuurd via de command line. Dit is heel handig als je het op een server moet draaien die ver weg staat.

Het commando om Icecast te starten is:

```
icecast -c icecast.xml
```

De truc is om het 'icecast.xml' bestand te vinden. Als je van bron geïnstalleerd hebt, zal je dit bestand in de Icecast2 bron map vinden in de conf map. Het bestand zal 'icecast.xml.dist' heten. Als het daar niet is: gebruik dan updatedb, en doe een locate of slocate om het bestand te vinden.

Dus laten we voorstellen dat ik in de Icecast2 bron map ben, en de 'icecast.xml.dist' bestand is in de 'conf' map, misschien is het handiger als ik hem naar mijn huidige werkmapp verplaats, en de naam verander naar 'icecast.xml'. Ik zal het cp commando gebruiken omdat ik dan een kopie van het origineel achterlaat (je weet maar nooit). Ik type daarom:

```
cp conf/icecast.xml.dist icecast.xml
```

nu kan ik het commando gebruiken:

```
icecast -c icecast.xml
```

Met geluk werkt het.. de -c parameter hier betekent 'config'. Het is een manier om het icecast commando te vertellen om het configuratiebestand 'icecast.xml' te gebruiken.

Als je merkt dat er een foutmelding is, ga dan naar het volgende onderdeel van icecast2 configureren. Als het werkt, hoef je alleen te weten dat het standaard wachtwoord hackme is.

Om MuSE te kunnen gebruiken met de Icecast server die je net hebt opgezet, moet je de volgende parameters invoeren (door het icecast.xml bestand aan te passen:

```
port : 8000  
password : hackme  
host : localhost
```

Gebruik dan welk mountpoint je maar wilt. /firststream zal bijvoorbeeld goed werken.

Als je dit doet als je de stream test, open dan xmms, en plaats de volgende URL in het 'open location' veld:

```
http://localhost:8000/firststream
```

je zou nu moeten horen wat er door je geluidskaart heenkomt.

Configuring Icecast

Icecast2 can be configured by editing the config file with a text editor. This document follows a certain syntax which is known as xml hence the Icecast2 configuration file should be named 'icecast.xml'. If you have a file called 'icecast.xml.dist' then this is the example configuration file that is installed with Icecast2. Rename the 'icecast.xml.dist' file to just 'icecast.xml'. Configuration File Format xml is a format that uses tags similar to HTML. The important thing to know is that each 'open tag' must have a 'close tag' otherwise the file will not be accepted by the server. A tag is something that is contained within an open and closed angle bracket like so:

```
<my-tag>
```

This example is called an 'open' tag, and each of these must have a corresponding closed tag. Closed tags start with a open angle bracket followed by a forward slash, like so:

```
</my-tag>
```

Hence a tag like " is the opening tag for the document, and the document is closed at the bottom of the file with ". Values can also be contained within tag pairs. these values are the crucial parts of the configuration file that you will change to customise the configuration of Icecast2 to your needs. To change values of the configuration file change the text within the xml tags. For example, if you need to change the source-password, you will change the entry from this:

```
<source-password>hackme</source-password>
```

To something like this:

```
<source-password>mypassword</source-password>
```

Comments

Comments are frequently used in the configuration file. Comments are simply remarks put in the file by the Icecast2 programmers to assist you in understanding the different sections of the configuration file. Each comment starts with ". This is the same syntax that HTML uses for comments. HTML Comments get ignored by the browser (they have no effect on the layout of a web-page), and consequently comments in this configuration file get ignored by the Icecast2 server. The below is an example of a comment you will find in the configuration file:

```
<!-- Uncomment this if you want directory listings -->
```

Changing the Config File

When you first start using Icecast2 there are just a few items within the configuration file that you need to worry about.

Authentication

```
<authentication>

    <!-- Sources log in with username 'source' -->
    <source-password>hackme</source-password>
    <!-- Relays log in username 'relay' -->

    <relay-password>hackme</relay-password>

    <!-- Admin logs in with the username given below -->
    <admin-user>admin</admin-user>
    <admin-password>hackme</admin-password>

</authentication>
```

The fields above control the login processes made by any type of software attempting to make a connection to the Icecast server. The exception is that media players (like Winamp / XMMS etc) do not need to have a login, they just connect without any necessary authentication. You only (for now) need to be concerned with the source-password section of this file. source-password is the password that encoders must use to make a connection to the Icecast server. In order to send a stream the encoder must first identify itself to the Icecast server by sending the correct password. If the password is correct then the server will accept the connection and you can start streaming. If the password is incorrect, or if there isn't a password sent by the encoder, then the Icecast2 server will not accept the connection and hence you will not be able to stream. There is only one password for encoding software, so if there are, for example, 10 people sending separate streams to the server then each must use the same password which is listed here. The default source-password for all **Icecast2** servers is "hackme" so its a good idea to change this to something else as a lot of people know this default.

Hostname and Port

You don't need to know what a hostname and port are for now, you just have to know what these fields should be set as.

```
<hostname>localhost</hostname>

<listen-socket>
    <port>8000</port>
    <!-- <bind-address>127.0.0.1</bind-address> -->
</listen-socket>
```

Every Linux machine has what is known as a hostname. To find out what this is simply type hostname into the terminal and enter the result into the field in this section. For example if I type hostname in the terminal and I get the result "darkstar.net" then my hostname field in this section would look like this:

```
<hostname>darkstar.net</hostname>
```

The default port for Icecast2 is "8000", just leave it as it is for now. Note: you will need to remember these two values (hostname and port) as they need to be put into the corresponding fields of your encoder.

Accounts Aanmaken

Software versie: **Icecast 2.3.1**

Icecast accounts worden aangemaakt door het configuratiebestand aan te passen met een tekstverwerkingsprogramma. Waarschijnlijk werk je met een config bestand op een server, zodat het nodig zal zijn om een tekstverwerker in de terminal te gebruiken. Probeer nano, pico, of emacs, vi, vim. Als je geen ervaring hebt met tekstverwerkers in de terminal, raad ik je de eerste twee aan.

Als je 1 wachtwoord wilt voor alle encorders (De standaard Icecast instelling) moet je naar het volgende gedeelte kijken:

```
<source-password>hackme</source-password>
```

In dit geval kan iedereen een stream maken, op ieder mountpoint van je server, met dit wachtwoord ('hackme'). Als je het wachtwoord wilt veranderen, hoef je alleen de tekst tussen de twee tags te veranderen in een nieuwe wachtwoord tekst.

Je kan echter ook verschillende mountpoints maken voor verschillende gebruikers, zodat ze met hun eigen wachtwoord moeten inloggen. Voor deze 'accounts' moet je aangeven wat het mountpoint is dat ze moeten gebruiken. De mogelijkheden om de parameters van deze mountpoints aan te passen zijn nogal uitgebreid, en je kunt het aantal luisteraars, de bitrate, waar de stream op wordt geslagen en zelfs een intro-file (een bestand dat wordt gespeeld voordat de luisteraar de stream hoort) instellen.

Bij dit type wordt ieder account getoond tussen de <mount></mount> tags. Als je een eenvoudig account wilt aanmaken, waar je iemand een gebruikersnaam en wachtwoord geeft om te kunnen streamen, is de configuratie die je toevoegt er zo uitzien:

```
<mount> <mount-name>newuser.mp3</mount-name> <password>hackmetoo</password> </mount>
```

Helaas is het zo dat als je mount 'accounts' zoals hierboven toevoegt aan het configuratiebestand, je Icecast moet dwingen om het configuratiebestand te lezen voordat de veranderingen door worden gevoerd. Je hoeft hier Icecast niet voor te herstarten, maar je moet wel een SIGHUP signaal sturen vanaf de command line. Je doet dit als volgt:

```
killall -HUP icecast
```

Je kunt ook enkele van de volgende parameters toevoegen (de mount-name is altijd nodig):

- mount-name
- username (gebruikersnaam)
- password (wachtwoord)
- max-listeners (maximale luisteraars)
- max-listener-duration (maximale luistertijd)
- dumpfile (dumpbestand)
- intro (intro bestand)

- fallback mount
- fallback-override
- fallback-when-full
- public
- stream-name (streamnaam)
- stream-description (stream beschrijving)
- stream-url (stream url)
- genre
- bitrate
- type
- subtype
- hidden (verborgen)
- burst-size
- mp3-metadata-interval
- authentication-type
- on-connect
- on-disconnect

Standaard Account Instellingen

Als we account instellingen zeggen, zijn dat eigenlijk de instellingen voor specifieke mountpoints. De volgende zijn de meest nuttige:

wachtwo0rd

Het wachtwoord dat nodig is voordat een stream geaccepteerd zal worden door de server op dat mountpoint. Het wachtwoord wordt ingevoerd in het wachtwoordveld van de encoder. Als de encoder een commandline encoder is, zal het wachtwoord of in de command line worden toegevoegd, of geplaatste in het config bestand. Als de encoder een GUI (grafische gebruikers interface) heeft, is er vaak een veldje waarin het wachtwoord ingevuld kan worden.

max-listeners (maximaal aantal luisteraars)

Het maximale aantal luisteraars dat een mountpoint tegelijkertijd kan ondersteunen. Als je de waarde op 50 zet bijvoorbeeld, kan je tot maximaal 50 gelijktijdige luisteraars hebben.

dump-file (dump bestand)

het **path** en de bestandsnaam waar de stream op dit mountpoint gearchiveerd (opgeslagen) moet worden.

intro

Dit geeft aan dat er een intro gestreamd moet worden op het mountpoint voordat de luisteraar de stream hoort. Als je bijvoorbeeld een live stream voor een evenement hebt, kan je een korte introductie afspelen als je luisteraar met de stream verbindt, waarna de stream volgt. In dit geval zou je deze optie gebruiken en een link naar het intro bestand invoegen.

Credits

Licence

All chapters copyright of the authors (see below). Unless otherwise stated all chapters in this manual licensed with **GNU Free Documentation License 1.2**

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts and no Back-Cover Texts. A copy of the license is included here:
<http://www.gnu.org/licenses/fdl.txt>

Authors

People that have contributed to this manual include (please add your name if you have made a contribution) :

BASIC CONFIG

© adam hyde 2006

ACCOUNTS

© adam hyde 2006

Modifications:

Lottes Meijer 2007

CREDITS

© adam hyde 2006, 2007

LINUX (SOURCE)

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

INTRODUCTION

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

LOGS (LINUX)

© adam hyde 2006

Modifications:

Lottes Meijer 2007

STARTING (LINUX)

© adam hyde 2006

Modifications:

Lottes Meijer 2007

WHAT IS STREAMING

© adam hyde 2005, 2006, 2007

Modifications:

Lotte Meijer 2007

Ruckert Martin 2006
Stella Brennan 2005

Free manuals for free software