IT Risk Management vid användning av Open Source Software

Kandidatuppsats 15 hp, kurs SYSK16 i Informatik

Författare: Axel Hellström
             Fredrik Moberg

Handledare: Björn Svensson

Examinatorer: Bo Andersson
               Magnus Wärja
IT Risk Management vid användning av Open Source Software

FÖRFATTARE: Axel Hellström och Fredrik Moberg

UTGIVARE: Institutionen för informatik, Ekonomihögskolan, Lunds universitet

FRAMLAGD: maj, 2018

DOKUMENTTYP: Kandidatuppsats

ANTAL SIDOR: 80

NYCKELORD: IT Risk Management, Open Source Software, Riskhantering

SAMMANFATTNING (MAX. 200 ORD):
Innehåll

1 Introduktion................................................................................................................. 1
  1.1 Bakgrund ............................................................................................................. 1
  1.2 Problemformulering............................................................................................ 3
  1.3 Forskningsfråga .................................................................................................. 3
  1.4 Syfte .................................................................................................................... 3
  1.5 Avgränsningar .................................................................................................... 4
2 Litteraturgenomgång ................................................................................................. 5
  2.1 Open Source Software ....................................................................................... 5
    2.1.1 Support .......................................................................................................... 5
    2.1.2 Licenser ......................................................................................................... 6
    2.1.3 Säkerhetsrisker ............................................................................................ 8
  2.2 IT Risk Management ......................................................................................... 8
    2.2.1 Riskstyrning (Risk Governance) ................................................................. 10
    2.2.2 Riskbedömning (Risk Evaluation) ............................................................... 11
    2.2.3 Riskhantering (Risk Treatment) .................................................................. 11
  2.3 Open Source & IT Risk Management ............................................................... 12
    2.3.1 Riskstyrning ................................................................................................. 12
    2.3.2 Riskbedömning............................................................................................ 12
    2.3.3 Riskhantering ............................................................................................. 13
  2.4 Litteratursammanfattning ............................................................................... 14
  2.5 Undersökningsmodell ...................................................................................... 15
3 Metod ....................................................................................................................... 17
  3.1 Datainsamling .................................................................................................... 17
    3.1.1 Intervjuobjekt .............................................................................................. 17
    3.1.2 Procedur ...................................................................................................... 18
  3.2 Transkribering & Analys ................................................................................ 19
  3.3 Validitet & Reliabilitet ..................................................................................... 19
  3.4 Etik ...................................................................................................................... 20
4 Resultat .................................................................................................................. 22
  4.1 Open Source Software ..................................................................................... 22
    4.1.1 Säkerhet ....................................................................................................... 22
    4.1.2 Support ........................................................................................................ 23
    4.1.3 Licenser ....................................................................................................... 24
    4.1.4 Resultattabell för Open Source Software ..................................................... 25
4.2 IT Risk Management .................................................................................. 26
  4.2.1 Riskstyrning ...................................................................................... 26
  4.2.2 Riskbedömning .................................................................................. 27
  4.2.3 Riskhantering .................................................................................... 28
  4.2.4 Resultattabell för IT Risk Management ............................................ 30
5 Diskussion ........................................................................................................ 31
  5.1 Open Source Software ........................................................................... 31
    5.1.1 Säkerhet ......................................................................................... 31
    5.1.2 Support .......................................................................................... 31
    5.1.3 Licenser ........................................................................................ 32
  5.2 IT Risk Management .............................................................................. 32
    5.2.1 Riskstyrning ................................................................................... 32
    5.2.2 Riskstyrning ................................................................................... 33
    5.2.3 Riskhantering ................................................................................ 33
6 Slutsats ............................................................................................................. 34
7 Referenser ........................................................................................................ 36
8 Appendix .......................................................................................................... 41
  8.1 Intervjumall ........................................................................................... 41
  8.2 Mailmall .................................................................................................. 42
  8.3 Intervjutranskriberingar ....................................................................... 43
    8.3.1 LDC .............................................................................................. 43
    8.3.2 Sony Mobile .................................................................................... 48
    8.3.3 CGI ............................................................................................... 57
    8.3.4 Prevas ............................................................................................ 62
    8.3.5 Organisation X ............................................................................... 66
Figurer

Figur 2.1: Skillnader mellan några av de vanligaste licenserna s. 7
Figur 2.2: ISACA IT Risk Framework s.10
Figur 2.3: Undersökningsmodell s.16
Tabeller

Tabell 2.1: Litteratursammanfattning s. 14-15
Tabell 3.1: Intervjuobjekt s.18
Tabell 4.1: Resultattabell för Open Source Software s.25
Tabell 4.2: Resultattabell för IT Risk Management s.29-30
1 Introduktion

1.1 Bakgrund

År 2016 omsatte mjukvaruindustrin 1038 miljarder euro (IDATE, 2016) och sysselsatte 6.7 miljoner människor i USA (CompTIA, 2016). Det är en global industri där mjukvara kan utvecklas tillsammans över hela världen och säljs över Internet, vilket skiljer den från många andra industrier (Buxmann, Diefenbach, Hess, 2013). Den ständiga transformeringen som sker i industrin möjliggör nya affärsmodeller och innovation som kan leda till signifikanta marknadsandelar och störningar på marknaden (Schief, 2014).

Ett resultat av den globala och föränderliga mjukvaruindustrin är fenomenet Open Source Software (OSS). Denna typ av mjukvara har tagit sig från små slutna kretsar inom hackerkulturen och universiteten till att förändra hela mjukvaruindustrin. Tidiga pionjärer inom området gav upphov till några av de mest använda programvarorna i dagens mjukvaruindustri. Idag är Linux det mest populära operativsystemet för webbservrar (W3Techs, 2018-a) och Wordpress det mest använda innehållshanteringssystemet (CMS), till exempel (W3Techs, 2018-b).


projektet när Linus Torvald skapade operativsystemkärnan Linux som i kombination med GNU-systemet utgjorde ett komplett operativsystem.


1.2 Problemformulering

Användningen av OSS är, som sagt, inte problemfri. En rapport från Black Duck Software (2017) visar att 96% av över 1000 kommersiella applikationer som undersöks, innehåller OSS-komponenter och att 67% av dem har säkerhetshål som, i genomsnitt, varit kända i över fyra år. Vidare är över 85% av applikationerna inte i överensstämmelse med licencerna för den OSS som används (Black Duck Software, 2017). Liknande resultat hittar vi i en undersökning från Snyk (2017) som visar att 77% av de 430 000 webapplikationerna som undersöks innehåller OSS-komponenter med erkända obehöriga funktioner och att över 85% av applikationerna inte är i överensstämmelse med OSS-licenserna.


Ett annat liknande problem är att OSS oftast inte erbjuder support på samma sätt som en mjukvaruleverantör gör. Enligt Tony Wasserman, professor vid Carnegie Mellon Silicon Valley och styrelseledamot i Open Source Initiative, finns bra support på olika OSS-forum men snabb support går inte att jämföra med ett supportsamtal till en mjukvaruleverantör. "People often respond very quickly to queries posted on the forum pages of widely-used open source projects, but that's not the same thing as a guaranteed vendor response in response to a toll-free telephone call." (Tony Wasserman, 2014).

Ef tersom en majoritet av organisationer arbetar med risk management (Deloitte, 2016) och mer specifikt ITRM (Ernst & Young, 2014) vet vi att stora resurser läggs på att reducera IT-risken. Trots detta så finns det problem som verkar förbises vid användandet av OSS. Om en organisation har en etablerad hantering av IT-relaterade risker borde dessa OSS-risker identifieras, bedömas, och hanteras. I denna uppsats är det därför av största betydelse att ställa teorin mot prakten genom att definiera olika litterära aspekter av ITRM som är relevanta när organisationer använder OSS. Därefter följer en datainsamling för att identifiera vilka aspekter av IT Risk Management som beaktas när organisationerna använder Open Source Software.

1.3 Forskningsfråga

Vilka aspekter av IT Risk Management beaktas när organisationer använder sig av Open Source Software?

1.4 Syfte

OSS är bevisligen en stor del av mjukvaruindustrin men innefattar också problem och risker. Därför har vi vetat att ITRM enligt teorin ska påverka organisationers hantering av IT-relaterade risker, vill vi undersöka om detta korrelerar med deras processer vid användningen av OSS.
1.5 Avgränsningar

Inom ITRM finns många olika aspekter att ta hänsyn till för att hantera IT-riser. Uppsatsens omfattning har därför avgränsats till de aspekter som är mest relevanta i förhållande till OSS. På samma sätt har inte alla risker och problem med OSS beaktats utan ett urval har gjorts utifrån vad som varit återkommande i litteraturen.
2 Litteraturgenomgång

Kapitel två ämnar att presentera uppsatsens undersökningsmodell samt den litteratur och forskning som ligger till grund för modellen. I kapitel 2.1 redogörs begreppet Open Source Software samt risker med OSS som varit återkommande i litteraturen. I kapitel 2.2 förklaras IT Risk Management och de aspekter som varit mest relevanta för vår forskningsfråga. I kapitel 2.3 presenteras litteratur som berör båda begreppen tillsammans.

2.1 Open Source Software


2.1.1 Support


nämns även upplärning av hur mjukvaran fungerar, vidareutbildning i effektivitet och automatisk versionsuppdatering som fler exempel på tjänster som ofta säljs av de företag som använder sig av freemium-modellen (Androutsellis-Theotokis et al. 2011; Buxmann et al. 2013).

Inom support talar man även om stöd från chefer och ledningsgrupper vid införande och användande av OSS (Back & Silic, 2017; Conradi et al. 2010). Om dessa nyckelgrupper inte förstår varför man använder OSS, vad det bär med sig för interna förändringar samt vilka möjligheter och risker som följer med OSS, kommer organisationen få problem (Back & Silic, 2017; Conradi et al. 2010). Det är också viktigt att vara uppmärksam på att OSS arbetar i enlighet med samt stödjer de strategier och mål som organisationen har (Conradi et al. 2010).

Under support kategoriseras även community, vilket handlar om hur uppdelningen av ansvar inom utvecklingen av OSS ter sig (Moroiu, Weiss & Zhao, 2006). De som utvecklar OSS gör det på sin fritid, ofta på grund av ett gemensamt intresse, en vision eller ett mål som delas mellan alla utvecklare inom projektet (Androutsellis-Theotokis et al. 2011; Moroiu et al. 2006).


2.1.2 Licenser


<table>
<thead>
<tr>
<th>Type of license</th>
<th>GPL</th>
<th>LGPL</th>
<th>MPL</th>
<th>BSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can be integrated into proprietary software and redistributed without an OS software license</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Modifications to OS licensed source code can remain proprietary on distribution</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Figur 2.1. Skillnader mellan några av de vanligaste licenserna (Buxmann et al., 2013, s. 193)


Mozilla Public License (MPL) är lik LGPL på det sättet att koden får användas och sedan säljas i proprietär mjukvara (Buxmann et al. 2013). Skillnaden ligger enligt Buxmann et al. (2013) i att MPL måste användas som den är, det vill säga inga modifieringar får göras på källkoden.

Om utvecklare inte är tillräckligt noggranna eller har några riktlinjer att följa då de väljer eller använder sig av OSS finns det stor risk att bryta många olika lagar gällande licenser (Back & Silic, 2017; Brown & Giera, 2004; Conradi et al. 2010). Ett problem som kan uppstå då man använder sig av OSS är risken att det i sig innehåller bitar eller hela segment av deriverad OSS, vilket i sig kan innehålla spår från ett annat OSS och så vidare (Back & Silic, 2017; Conradi et al. 2010).

2.1.3 Säkerhetsrisker

Det råder konstant diskussion mellan de som förespråkar proprietär mjukvara och de som förespråkar OSS och det blir aldrig någon klargörelse kring vilken typ av mjukvara som är säkrast (Back & Silic, 2017; Booch & Brown, 2002; Brown & Giera, 2004; Schryen, 2011).

De mest förekommande ämnena som brukar diskuteras är kontroll av koden samt kontroll över utvecklare (Booch & Brown, 2002; Payne, 2002; Schryen, 2011). De som säger att OSS är mindre säker än proprietär mjukvara argumenterar att när flera olika utvecklare med dold agenda sitter och skriver koden till ett program finns det risk att någon lägger in en bakdörr eller implementerar övervakning i programmet som sedan kan utnyttjas (Back & Silic, 2017; Booch & Brown, 2002; Payne, 2002; Schryen, 2011). Motargument mot detta är att det är mycket svårt att implementera farlig kod då all kod granskas av andra utvecklare innan den implementeras i en version som skickas ut till användare (Booch & Brown, 2002; Payne, 2002). Vidare argumenterar de som är för OSS att proprietär mjukvara kan vara farligare då man som användare inte vet vad som händer bakom kulissen när det gäller mjukvaran man arbetar med samt ger information till, eller hur många som har sett över säkerheten (Booch & Brown, 2002; Payne, 2002).


2.2 IT Risk Management

Enligt ISO (2009, s. 2) definieras risk management som “[the] coordinated activities to direct and control an organization with regard to risk”. Risk management utgörs av risk management-processen:

“[The risk management process is the] systematic application of management policies, procedures and practices to the activities of communicating, consulting, establishing the context, and identifying, analyzing, evaluating, treating, monitoring and reviewing risk.” (ISO, 2009, s.2)


ITRM är en komponent av ett större risk management-system och en del av risk management-processen (ISACA, 2009). ISACA (2009, s. 7) definierar ITRM som: “the business risk associated with the use, ownership, operation, involvement, influence and adoption of IT within an enterprise or organization”. ITRM omfattar inte bara IT-risker som kan ha en negativ inverkan på organisationen, utan också förbättra effektiviteten i verksamhetsprocesser, eller möjliggöra nya affärsinitiativ (ISACA, 2009). Enligt NIST (2012) berör ITRM många olika roller i en organisation, som till exempel företagsledningen, informationsägare, processägare, IT-inköpare, och informationssäkerhetsansvariga.


1. ISO 27005 (2008)
2. ISACA Risk IT Framework (2009)
3. OCTAVE (Caralli, Stevens, Young, Wilson, 2007)
4. NIST (2012)

2.2.1 Riskstyrning (Risk Governance)

IT Risk Management vid användning av Open Source Software

Axel Hellström & Fredrik Moberg

ITRM bör integreras med ett existerande enterprise risk management-system för att få en holistisk syn på risk management-processen (Caralli et al. 2007; ISACA, 2009).

2.2.2 Riskbedömning (Risk Evaluation)


Konsekvens- och sannolikhetsvärdena räknas ut med hjälp av en formel för att mäta IT-risker (Caballero, 2009). Det finns många olika formler för att mäta IT-risker men en vedertagen sådan är:

\[ \text{Risk} = \text{Asset} \times \text{Threat} \times \text{Vulnerability} \] (Caballero, 2009; Carr, Rainer, Snyder, 1991)

För att mäta risker måste organisationen först identifiera sårbarheter (vulnerabilities) i tillgångar (assets) som måste skyddas mot hot (threats) (Caballero, 2009). Många har kritiserat formulan för att vara för generell och ej matematiskt gångbar vilket resulterat i många olika revideringar (ISACA, 2009; Katsikas, 2013; Kouns & Minoli, 2010).

2.2.3 Riskhantering (Risk Treatment)


2.3 Open Source & IT Risk Management

Det finns mycket litteratur och forskning kring möjligheterna och de positiva sidorna av OSS, medan riskerna och de negativa sidorna inte är lika väl undersökt och förstådda (Back & Silic, 2015; Conradi, Cruzes, Hauge, Skarpenes & Velle, 2010)

Ett av de vanligaste misstagen när organisationer implementerar OSS är bristfällig applicering av risk management (Franch et al., 2013; Franch & Susi, 2016). En holistisk risk management-process bidrar till att både tekniska och verksamhetsstrategiska OSS-relaterade risker kan identifieras, bedömas och hanteras (Back & Silic, 2015; Franch & Susi, 2016).

2.3.1 Riskstyrning


2.3.2 Riskbedömning

Det finns stora mängder öppen data (källkod och community-relaterade artefakter) om olika OSS-projekt som kan användas för att stödja, förenkla och förbättra risk management-processen (Bai, Kenett, Yahav, 2014; Franch, Kenett, Susi, Galanis, Glott, Mancinelli, 2015). Datan bör användas till att identifiera både tekniska och strategiska risker (Bai et al. 2014;


Dessa aspekter är bland andra:

- Projektets livslängd (age)
- Senaste utgåvan (release)
- Popularitet
- Dokumentation & Support
- Kod- och arkitekturkvalitet
- Licenser

(Woods & Guliani, 2005)


2.3.3 Riskhantering

Riskhantering inom OSS handlar om att reducera och hantera riskerna som identifierats och analyserats/bedömts (Mestre, Rodrigues & Soares, 2011). Det är inte alltid det går att eliminera eller ens reducera en risk eftersom det ligger utanför organisationens domän (Conradi et al., 2010). Dock är de två tidigare aktiviteterna (riskidentifiering och riskbedömning) en sorts riskhantering i sig eftersom det bidrar till medvetenhet och utvärdering av risker (ISACA, 2009).

Några generella framgångsfaktorer som kan reducera OSS-relaterade risker är följande:

- Anställdas attityd, medvetenhet, och kompetens (Bonaccorsi, Giannangeli & Rossi, 2006; Exton, Fitzgerald & Glynn, 2005; Gomez & Moreno, 2012)
- Tillgång till support (Fitzgerald, 2009)
- Andra framgångsexcempel (Ågerfalk, Deverell, Fitzgerald & Morgan, 2005)
- Ingen inläsning (Goode, 2005)
- OSS-policy (Woods & Guliani, 2005)

Ytterligare en faktor som kan reducera OSS-relaterade risker är att ta fram en policy som beskriver hur OSS ska användas i organisationen (Woods & Guliani, 2005). Denna kan till exempel innehålla en lista över vilka licenser som får användas för att undvika att intellectual property (IP) läcker ut (Woods & Guliani, 2005).

I en undersökning gjord på Telenor av Conradi et al. (2010) identifierades följande steg som mest vitala av Telenors anställda vid användning av OSS:

- Öka de anställdas kunskap om OSS (antingen genom att anställa eller utbilda)
- Öka de anställdas medvetenhet och inställning om pågående OSS-relaterade implementeringar, initiativ och projekt
- Förankra ledningens engagemang och stöd för OSS
- Ha tillgång till support

### 2.4 Litteratursammanfattning

<table>
<thead>
<tr>
<th>Kategori</th>
<th>Sammanfattning/Innehåll</th>
<th>Litteratur</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licenser</td>
<td>• Typer av licenser  &lt;br&gt;  o GPL  &lt;br&gt;  o LGPL  &lt;br&gt;  o MPL  &lt;br&gt;  o Apache  &lt;br&gt;  • Risk för licensbrott  &lt;br&gt;  • Deriverade licenser</td>
<td>• Androutsellis-Theotokis et al. (2011)&lt;br&gt;• Back &amp; Silic (2017)&lt;br&gt;• Brown &amp; Giera (2004)&lt;br&gt;• Buxmann et al. (2013)&lt;br&gt;• Conradi et al. (2010)</td>
</tr>
</tbody>
</table>
### Tabell 2.1: Litteratsammanfattning

<table>
<thead>
<tr>
<th>Riskstyrning</th>
<th>Stöd från ledningen</th>
<th>Definiera roller och ansvar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Belle et al. (2006)</td>
<td>Caralli et al. (2007)</td>
</tr>
<tr>
<td></td>
<td>Woods &amp; Guliani (2005)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Riskbedömning</th>
<th>Identifiera data om:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Tillgångar</td>
</tr>
<tr>
<td></td>
<td>Hot</td>
</tr>
<tr>
<td></td>
<td>Sårbarheter</td>
</tr>
<tr>
<td></td>
<td>Konsekvenser</td>
</tr>
<tr>
<td></td>
<td>Mått IT-risk (R = T * V * A)</td>
</tr>
<tr>
<td></td>
<td>Analysera datan</td>
</tr>
<tr>
<td></td>
<td>Kvantitativt</td>
</tr>
<tr>
<td></td>
<td>Kvalitativt</td>
</tr>
<tr>
<td></td>
<td>OSS-riskverktyg</td>
</tr>
<tr>
<td></td>
<td>Open Source Maturity Model</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Riskhantering</th>
<th>Hantera risk genom att:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Undvikta risk</td>
</tr>
<tr>
<td></td>
<td>Reducera risk</td>
</tr>
<tr>
<td></td>
<td>Överföra risk</td>
</tr>
<tr>
<td></td>
<td>Acceptera risk</td>
</tr>
<tr>
<td></td>
<td>Faktorer som kan reducera OSS-riser:</td>
</tr>
<tr>
<td></td>
<td>Anställdas attityd, medvetenhet, och kompetens</td>
</tr>
<tr>
<td></td>
<td>Tillgång till support</td>
</tr>
<tr>
<td></td>
<td>Andra framgångse exempel</td>
</tr>
<tr>
<td></td>
<td>Ingen inläsning</td>
</tr>
<tr>
<td></td>
<td>OSS-policy</td>
</tr>
</tbody>
</table>

|              | Bonaccorsi et. al. (2006) |
|              | Exton et. al. (2005)       |
|              | Fitzgerald (2009)          |
|              | Goode (2005)               |
|              | ISACA (2009)               |
|              | ISO (2008)                 |
|              | NIST (2012)                |
|              | Woods & Guliani (2005)     |
|              | Ågerfalk et. al. (2005)    |

### 2.5 Undersökningsmodell

Undersökningsmodellen i figur 2.3 baseras på tidigare forskning och litteratur som sammanställts i detta kapitel (2). Vidare ligger undersökningsmodellen till grund för datainsamlingen och för att besvara forskningsfrågan. De huvudsakliga områdena vi identifierat inom ITRM är riskstyrning, riskbedömning, och riskhantering. Som tabell 2.1 visar finns det många olika aspekter inom huvudområdena och det är dessa aspekter vi syftar på i vår forskningsfråga. Vad gäller OSS har vi identifierat säkerhet, support och licenser som vanligt förekommande risiker och problem enligt litteraturen. Dessa risker tillsammans med ITRM-aspekterna ligger till grund för våra intervjufrågor och därmed datainsamlingen.
Figur 2.3: Undersökningsmodell
3 Metod

För att kunna ge svar på frågeställningen krävs en god inblick i hur både val av mjukvara går till samt hur riskhantering ter sig inom en organisation. Då kriterierna inte uppfylls av många anställda inom en organisation, om ens någon, förstod vi i ett tidigt skede att vi var ute efter kvalitativ data och ej kvantitativ. Vi visste även att ämnets bredd skulle innebära att en intervjumall kunde bli svår att följa, trots detta valde vi att använda oss av en mall då detta enligt Backman (2016) är ett bra instrument för att kunna redovisa replikation och evaluering.

3.1 Datainsamling

Backman (2016) beskriver kvantitativ metodik som användningen av mätningar, matematik och kvantifiering som utmynnar i numeriska observationer och resultat. Kvalitativ metodik använder sig inte av siffror utan verbala formuleringar (Backman, 2016). Därav föll den kvalitativa metodiken mer i linje med det arbete vi valde att genomföra för att införska data.

3.1.1 Intervjuobjekt

Vi fick kontakt med alla intervjuobjekt genom att maila direkt till de organisationer vi ville intervjua, oftast genom en generell info-mailadress som gick till deras kundtjänst eller service desk. I mailen uttryckte vi att vi ville bli vidarebefordrade till någon de tror skulle kunna hjälpa oss. De intervjuobjekt vi ämnade få kontakt med var de som innehör roller som vi förstod har hög förståelse för eller arbetar med både övergripande systemarkitektur och internt säkerhetsarbete (Appendix 8.2). Vi ville ha kontakt med dessa roller då vi antog att de kunde ha insyn i hur både ITRM och OSS fungerar och används inom deras organisationer. Utöver detta var kravet även att organisationen i fråga antingen använder sig av OSS eller aktivt har valt att inte använda det. Vi tänkte att en organisation som medvetet valt bort OSS-alternativ förmodligen har genomfört någon slags riskanalys och skulle således kunna ge svar på frågeställningen. Vi valde främst att fokusera på organisationer inom IT-branschen, helst de som producerar IT-tjänster eller mjukvara, då vi antog att dessa organisationer skulle ha högst chans att kunna erbjuda intervjuobjekt.

Vi erbjöd alla intervjuobjekt möjligheten att vara anonyma. Vi förstod att detta var viktigt då uppsatsens ämne till viss del hanterar väsentliga delar av en organisations säkerhet. För att garantera en ömsesidig bild av anonymitet skickade vi alltid vår pseudonyma beskrivning av organisationen samt transkriberingen av intervjun till intervjuobjektet för godkännande innan vi använde data från intervjun.

Totalt intervjuade vi fem företag med spridda arbetsområden och vitt differerande tjänster och produkter inom IT-branschen, vilka syns i tabell 3.1. En organisation ville vara anonym och där har vi ersatt organisationens namn med “Organisation X” och intervjupersonen med “Person X”.
3.1.2 Procedur


Syftet med frågornas upplägg var att först ge intervjuobjektet en bild om vilka vi var, vår bakgrund samt varför vi skriver uppsatsen. Sedan ville vi få intervjuobjektet att börja prata lite och bli varm i kläderna genom att svara på enklare frågor om organisationen och vilket arbete intervjuobjektet genomför. Vi ville även att intervjuobjektet skulle presentera sin uppfattning av de två huvudsakliga begreppen som uppsatsen undersöker, ITRM och OSS. Om det skiljde
sig från hur vi har definierat det i det teoretiska kapitlet, förklarade vi vår definition för att försäkra oss om att vi pratade om dessa ämnen på samma nivå.

3.2 Transkribering & Analys

Det bestämdes vid ett tidigt skede att transkriberingen endast skulle innehålla det som var av relevans för intervjun och ämnet. Frågor eller utsvävningar som inte på något sätt berörde frågeställningen eller intervjufrågorna som är nedskrivna i intervjuuppgifterna (Appendix 8.2) togs därför inte med i transkriberingen. Tillfälliga stämningar, utfyllnadssord som "ehh" och lång betänkhetid är några fenomen som vi valde att inte ta med i transkriberingen då vi anser att det inte påverkar svaren samt underlättar läsbarheten. Om en ljudinspelning skulle gå förlorad var tanken att intervjun skulle återges så gott som möjligt, men inte ordagrant, baserat på de anteckningarna som fördes under intervjun.


3.3 Validitet & Reliabilitet


Den externa validiteten handlar om i vilken grad resultatet av en undersökning kan generaliseras (Jacobsen, 2002). Detta är svårt när det kommer till en kvalitativ ansats då målet inte är att generalisera i större sammanhang (Jacobsen, 2002). Enligt Jacobsen (2002) är målet att utveckla en mer generell teori utifrån den data som erhålls från ett mindre urval av
intervjuobjekt. Detta faller i linje med den väg vi valde att följa gällande den externa
giltigheten då ämnet är av specifik natur och inte något som enkelt kan generaliseras till en
större mängd.

Reliabiliteten kretsar kring ifall det är något i själva undersökningen som har påverkat
resultatet av denna (Jacobsen, 2002). En av dessa effekter som Jacobsen (2002) skriver om är
kontexteffekten, den effekt som påverkar hur intervjuobjektet kan bete sig beroende på hur
bekväm den är i sin kontext, det vill säga hur naturlig omgivningen är. En naturlig omgivning
can enligt Jacobsen (2002) vara i intervjuobjektets hem eller på dess arbetsplats där den
befinner sig dagligen och inte känner sig obekväm eller hotad. Vi valde att utföra intervjuerna
på intervjuobjektets arbetsplats eller via videomöte när den befann sig på sin arbetsplats för
att minimera externa påverkningar via omgivning. Utöver detta bad vi intervjuobjektet själv
bestämma en tid då det passade, samt att intervjun skulle ta cirka 30 minuter (Appendix 8.2)
för att ge en bild av hur stor del av arbetsdagen intervjun i sig skulle kräva. Detta faller i linje
med det Jacobsen (2002) nämner gällande planerad undersökning. Om man vill ha planerade
och genomtänkta synpunkter lämpar det sig att använda en planerad undersökning, medan en
oplanerad undersökning ger större chans till spontana äsikter och känslor (Jacobsen, 2002).

3.4 Etik

Enligt Jacobsen (2002) finns det tre huvudsakliga grundkrav som en undersökning måste
uppfylla för att vara etisk försvarbar: informerat samtycke, rätt till privatliv och korrekt
återgivning.

Informerat samtycke innefattas av flera mindre krav som bör försöka uppfyllas så gott som
möjligt (Jacobsen, 2002). Dels måste de som undersöks fritt få välja om de ska delta i
undersökningen eller ej, vilket kan tyckas verka löjligt, men det är viktigt för den insamlade
datan att det inte finns någon bakomliggande orsak eller påtryck från en extern aktör att
personen ska delta i undersökningen (Jacobsen, 2002). Dessutom måste de som undersöks
enligt Jacobsen (2002) få full tillgång till information kring syftet med undersökningen samt
information om eventuella fördelar och nackdelar med sitt deltagande.

Rätt till privatliv syftar till att beakta information och uppgifter rörande privatlivet hos den
som blir undersökt (Jacobsen, 2002). Det är därför av stor vikt att eftersträva diskretion vilket
uppnås genom att erbjuda deltagarna garanterad anonymitet och konfidentialitet, för att
undvika att informationen kan bli kopplad till en privatperson (Jacobsen, 2002).

Det sista kravet är enligt Jacobsen (2002) krav på riktigt presentation av data. Detta innebär
att den intervjuade personen ska bli korrekt återgiven och i rätt sammanhang (Jacobsen,
2002). För att se till att detta sköts på ett godtycklig vis kan uppgiftslämnaren alltid kräva
fullständig återgivning (Jacobsen, 2002).

Vi har försökt uppnå alla dessa krav genom följande tillvägagångssätt: Vi lät organisationerna
få full information gällande uppsatsens syfte och möjliga frågor vid första kontakt, vi lät även
organisationerna själva hänvisa till de personer som de trodde skulle kunna hjälpa oss, det vill
säga de som besatt den lämpliga kompetensen. Vi frågade alla intervjuobjekt huruvida de ville
vara anonyma eller ej samt ifall de godkände en ljudinspelning av intervjun. Till sist skickade
vi fullständig transkribering för godkännande samt korrigerings till varje intervjuobjekt för att försäkra oss om att båda parter var eniga i att datan som presenteras är korrekt.
4 Resultat

Nedan presenteras resultatet av den empiriska undersökningen. Intervjuerna är sammanställda och kategoriserade enligt huvudområdena i undersökningsmodellen.

4.1 Open Source Software

Vår definition av begreppet Open Source Software (OSS) är enligt intervjunallen (Appendix 8.2) källkod som är fullt tillgänglig för andra att modifiera och distribuera utan begränsningar. Under alla intervjuer definierade intervjuobjekten begreppet snarlikt. Philip Vendil beskrev det följande:


4.1.1 Säkerhet

De svaren som presenteras nedan dök upp i samband med att vi frågade om intervjuobjekten har stött på några säkerhetsproblem med OSS. Både CGI och Sony Mobile sade att de proaktivt arbetar för att undvika dessa risker. CGI förklarade att på grund av den utveckling som har skett inom säkerhet har det blivit lättare att undvika deriverade sårbarheter:

“Där har det ju blivit mycket bättre på senare tid tycker jag, med att få säkerhetsvarningar i bibliotek. Tidigare visste man inte alls, då var det väldigt svårt att hålla reda på för att, det är ju liksom beroenden som av OS-bibliotek så det kan vara hundratals bibliotek så det är svårt att veta om den där längst ner på listan har fått en sårbarhet” (Philip Vendil, 2018).

CGI försvarar sig mot detta genom att genomföra kontroller av all kod de väljer att implementera.

Sony Mobile menade att OSS är säkrare än proprietär mjukvara och att den åsikten delas av många andra som använder sig av OSS:


Men trots det så innebär säkrare inte helt och hållet säkert enligt Sony Mobile och att man hela tiden måste vara uppmärksam på vad man tar in för komponenter. Även fast de har en god bild av säkerheten inom OSS så granskar de fortfarande all kod de tar in.
Organisation X har en intern process som utför kontroll av koden de använder. Organisation X belyste också att det är viktigt att kontinuerligt kontrollera dessa komponenter:


Prevas nämnde inga konkreta säkerhetshål eller risker men uttryckte en stark oro för mer traditionella företag som ska förnya sina produkter eller maskiner med hjälp av Internet of Things. Inom Internet of Things används det enligt Prevas mycket OSS och när detta sätts i samband med verksamheter som tidigare inte jobbat med denna typ av teknologi finns det stora risker att OSS kan utnyttjas för att skada verksamheten.

LDC svarade att de enbart använder sig av större och mer beprövade typer av OSS och lägger därav ingen tid eller kraft på att undra om mjukvaran är säker, de litar dom på att den är.

4.1.2 Support

Angående support var majoriteten positiv till den support de anser att man får kring OSS. CGI sade att supporten hos proprietär mjukvara kan vara krångligt då man ofta måste skapa en support issue och sedan få vänta tills någon på företaget med rätt kompetens kan erbjuda hjälp. När det gällde supporten hos OSS tyckte CGI istället att det oftast fungerar bättre än proprietär mjukvara:


Även Prevas höll med om god support baserat på tidigare erfarenheter med OSS där communityn är stor:


Till skillnad från Prevas och CGI påpekade LDC att support inom OSS ibland kan vara bristfällig om de behöver en förändring då det inte finns några krav eller incitament för utvecklarna att lösa de problem LDC stöter på vid användningen av mjukvaran. Men LDC var inte bara negativ i sin syn på supporten kring OSS utan påpekade även att det finns bra communitys som har stort fokus på användarna och till och med erbjuder tränings och utbildning inom mjukvaran:

“Min kollega Peter som jobbar mest med implementationen tekniskt ska nu gå på kurs i Elastic Stack i Schweiz, vilket visar på att communityn är stor och det finns stora och bra möjligheter att utbilda sig.” (Magnus Persson, 2018).
Sony Mobile nämnde att supporten kan skifta, men att de oftast får bra respons när de väl kontakter utvecklarna. Eftersom Sony Mobile anser sig själva vara stora när det kommer till användning av OSS brukar utvecklare bli positivt överraskade när någon från Sony Mobile ber dom att ändra någonting vilket gör att de ofta får bra hjälp.

Organisation X sade att dom för närvarande fortfarande använder sig av många stora proprietära lösningar men att de sakta men säkert börjar använda mer och mer OSS. Därför visste Person X inte särskilt mycket gällande support när det kom till deras användning.

4.1.3 Licenser

Vid frågor kring licenser svarade en stark majoritet att de arbetar för att undvika licensbrott så gott som möjligt. CGI beskrev att de har en central policy som konstaterar exakt vilka licenser man får eller inte får använda och sedan är det upp till utvecklarna att se till att följa policyn då de väljer en OSS-komponent. Till exempel får de inte använda sig av GPL då det finns risk att CGIs kunder skulle behöva släppa sin mjukvara som OSS.

Prevas förklarade hur de hade gjort på tidigare projekt där kunder behövde hjälp med att genom söka och kategorisera alla licenser de använde sig av:

“[...] vad vi gjorde var egentligen en gap-analys på vilka OS-paket hade man och hur hade man behandlat deras licenser. [...] Det var nog så att det blev en liten sån aha-upplevelse för kunden att, okej var det så mycket att göra, det hade vi inte riktigt räknat med.” (Håkan Erlandsson, 2018).

Organisation X sade att de inom koncernen har en avdelning som varje intern verksamhet kan använda som en tjänst för att kontrollera att inga licensbrott sker:

“Ja, vi har inom koncernen en central hantering av Software Asset Management och licenser [...] då köper vi kontrollen av det som tjänst så att säga.” (Person X, 2018).

Gällande licenser så svarade LDC till skillnad från de andra att de inte aktivt undersöker vilken licens det är på grund av hur dom som statlig myndighet använder sig av Open Source. Sedan förklarade LDC att de inte tror att någon som använder sig av OSS undersöker licenser över huvud taget, baserat på hur dom själva använder sig av det:


Sony Mobile sade att de använder sig av sin juridiska avdelning för att se till att alla komponenter de tar in och använder sig av har en licens som de kan och får använda sig av:

Sony Mobile nämnade även att det är viktigt med strukturerade processer och att deras Head of Open Source måste se över alla val av nya OS-komponenter som görs inom verksamheten:

“Vårt bästa försvar mot sådana här copyright breach of license är att se till att vi har processer på plats, och det har vi. [...] När vi väljer att ta in Open Source görs en Open Source-evaluation som skickas till mig. Sen får legal godkänna licensen [...]” (Carl-Erik Mols, 2018).

4.1.4 Resultattabell för Open Source Software

<table>
<thead>
<tr>
<th>Organisation</th>
<th>Säkerhet</th>
<th>Support</th>
<th>Licenser</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sony Mobile</td>
<td>Tycker OSS är säkrare än proprietärt men menar att det ändå finns säkerhetsrisker.</td>
<td>Supporten kan skifta. Oftast får de bra respons från communityn eftersom Sony har bra rykte i OSS-communityt.</td>
<td>Legal-avdelningen sköter licenser</td>
</tr>
<tr>
<td>CGI</td>
<td>Tycker att det blivit bättre på senare tid. Ser biblioteksberoenden som ett stort problem.</td>
<td>Tycker att supporten hos OSS-communityn kan vara bättre än proprietär.</td>
<td>Följer CGI:s licenspolicy</td>
</tr>
<tr>
<td>Prevas</td>
<td>Uttryckte oro inom IoT.</td>
<td>Tycker att supporten är bra i stora communities.</td>
<td>Gör GAP-analyser för att genomsöka och kategorisera licenser</td>
</tr>
<tr>
<td>Organisation X</td>
<td>Nämnde inga specifika problem.</td>
<td>Använder mycket proprietära lösningar och visste därför inte särskilt mycket gällande OSS-support</td>
<td>Software Asset Management-avdelningen sköter licenser</td>
</tr>
</tbody>
</table>

Tabell 4.1: Resultattabell för Open Source Software
4.2 IT Risk Management

Överlag var de flesta intervjuuppsåten mer involverade och kunniga inom OSS och utveckling än ITRM. CGI, LDC och Sony Mobile nämnde att deras organisationer arbetar efter ITRM-ramverk, mer specifikt ISO 27000-serien (LDC & CGI) och NIST (Sony Mobile). Dock var detta något som hanterades ur ett bredare verksamhetsperspektiv och därför inte märktes av speciellt mycket i det dagliga arbetet hos CGI och LDC. På frågan om organisationen använder något ITRM-ramverk svarade CGI:

“Nej, alltså jag vet ju att CGI globalt jobbar mot ISO men det är inget jag har varit i kontakt med under utveckling eller sådär.” (Philip Vendil, 2018)

Sony Mobile svarade att ITRM hanteras av Software Security-avdelningen men att risktänket även finns inom OSS:

“Ja, nu sitter jag ju omgiven av mina kollegor som tillhör software security [...]. Lika mycket systematik som de har inom security har jag inom Open Source.” (Carl-Eric Mols, 2018)

Person X var mer involverad i ITRM-processen och förklarade att den genomsyrar allt de gör:

“[...] vi har en funktion i mitt team som kallas precis IT Risk Management [...]. Vi jobbar både proaktiv och reaktivt med riskhantering och omsätter det mesta, alla förändringar vi gör går igenom vår riskprocess och vi jobbar oftast, i alla fall när det kommer jobb, riskbaserat.” (Person X, 2018)

4.2.1 Riskstyrning

Vid frågan gällande hur väl informerade ledningen är om OSS-risker var svaren blandade. Sony Mobile, CGI och Organisation X svarade alla att ledningen hade god förståelse för riskerna och möjligheterna kring användandet av OSS. Både Sony Mobile och CGI nämnade att ledningen varit med och tagit fram en OSS-policy som beskriver hur OSS ska användas. Organisation X arbetar med riskstyrning på ett bredare verksamhetsplan som innefattar alla risker:


Prevas svarade att de har sett det mesta när de är ute och besöker kunder och att mycket av det handlar om bakgrunden hos ledningen. Om en majoritet av ledningen har teknisk bakgrund eller god förståelse för IT går arbetet med risker kring OSS ofta bra, annars finns det en risk för frktion och att riskerna inte riktigt når fram.

LDC svarade att de ibland kan ha svårt att förklara riskerna och få ledningen medvetna om varför de måste åtgärdas proaktivt. Problemet ligger enligt LDC i att få ledningen att förstå varför man ska ta beslut som kostar pengar nu för att möjligtvis få en vinst i framtiden. Speciellt blir det svårt då de inte har några säkerhetskrav på verksamheten:
“I vår organisation finns inget krav att man ska ha en viss säkerhet, förutom att vi ska försöka använda ISO27005.” (Magnus Persson, 2018).

4.2.2 Riskbedömning

Samtliga intervjuupptag svarade att de genomför någon typ av riskbedömning av OSS-komponenterna de använder men tillvägagångssättet varierar. Både Sony Mobile, CGI, och Prevas använder automatiserade verktyg som upptäcker risker och problem med OSS-komponenter:


Pluginet som Vendil nämner används för att kolla upp sårbarheter i OSS-sårbarhetsdatabaser.

Sony Mobile använder liknande verktyg men mer av en automatiserad helhetslösning:


För Prevas beror riskbedömningen på deras kunders krav men, likt Sony Mobile, har de också använt Black Duck:

“Det försöker vi göra i den mån vi får lov av kunden, eller får tid eller där vi kan övertyga kunden om att ja men det är nog rätt vettigt att göra det. […] Men vi försöker ta reda på för kunden, är ni medvetna, har ni gjort någon riskanalys, är ni medvetna om vad som händer, har ni vägt in dom parametrarna i valet? […] bland annat finns det verktyg för det, till exempel ett företag som heter Black Duck […] och det har jag varit med om att vi har använt.” (Håkan Erlandsson, 2018).

LDC och Organisation X använder inga verktyg men har manuella processer på plats för att godkänna OSS och identifiera risker. LDC gör en enkel riskbedömning genom att kolla på OSS-specifika faktorer som de anser viktiga:

“Vi kollar på communityn, produkten, livslängd, patchfrekvens för att försöka avgöra om det är värt det eller inte.” (Magnus Persson, 2018).

Organisation X har en godkännande process för OSS-komponenter som görs av vissa IT-arkitekter och säkerhetsfunktionen:
“Det kräver ändå en extra sväng av godkännande och utlåtande från dom som vill, ja, dom inom IT-organisationen framförallt vår Enterprise-arkitekt och dess nätverk av IT-arkitekter. Men också av våran säkerhetsfunktion, för att dels kunna säga att jo men den här är i sitt nuvarande stadie okej men också för att få till en kontinuerlig bevakning över OS-komponenter.”
(Person X, 2018)

På frågan om organisationerna använder någon formel $(T \times V \times A = R)$ för att beräkna IT-risker hade samtliga intervjupersoner hört talas om den men kunskapen var inte speciellt djup. LDC tyckte att det var svårt att beräkna risking på det sättet men att formeln hade ett teoretiskt värde. Sony Mobile, CGI, och Organisation X hade alla sin egna versioner av formeln.

4.2.3 Riskhantering

Då riskhanteringen är specifik beroende på vilken risk det gäller ser svaren ganska olika inom detta område. LDC mitigerar OSS-risker genom att välja OSS-komponenter som är uppbackade av ett aktivt och stort community, samt genom att undvika OSS i sina huvudplattformar:

“Vi väljer inga små konstiga OSS då vi måste kunna garantera drift på ett eller annat sätt [...] Problemet ligger att vi måste ha leverantörer att skylla på/peka på om något går snett. Gällande till exempel huvudplattformarna måste vi därför ha en proprietär leverantör för att kunna peka på vid diverse problem.” (Magnus Persson, 2018).

CGI har en kontinuerlig riskhanteringsprocess där de identifierar och analyserar risker för att sedan åtgärda dem:


Organisation X säger följande om riskhantering på ett generellt plan:

“Vi jobbar både proaktivt och reaktivt med riskhantering och omsätter det mesta, alla förändringar vi gör går igenom våran riskprocess och vi jobbar oftast, i alla fall när det kommer jobb, riskbaserat.” (Person X, 2018).

Vidare förklarar Person X att efter riskbedömning kan riskerna hanteras på olika sätt:


Sony Mobile:s riskhanteringsprocess involverar den juridiska avdelningen och Intellectual Property-avdelningen:

Vidare påpekar Mols att kulturen och medvetenheten hos de anställda är det bästa försvar mot risker:

### 4.2.4 Resultattabell för IT Risk Management

<table>
<thead>
<tr>
<th></th>
<th>Riskstyrning</th>
<th>Riskbedömning</th>
<th>Riskhantering</th>
</tr>
</thead>
</table>

Tabell 4.2: Resultattabell för IT Risk Management
5 Diskussion

I kapitel fem analyseras och diskuteras datan från den empiriska undersökningen ytterligare och ställs mot tidigare forskning och litteratur som presenterats i kapitel två. Diskussionen är, likt tidigare, kategoriserad efter undersökningsmodellens huvudområden.

5.1 Open Source Software

Datainsamlingen gav bilden att intervjuobjekten till stor del var väldigt positiva till OSS och att användningen uteslutande hade givit mer fördelar än nackdelar.

5.1.1 Säkerhet


5.1.2 Support


5.1.3 Licenser

Vid datainsamlingen gällande licenser visade de flesta organisationer på ett väletablerat och moget arbete som behandlade kontroll av licenser centrat inom organisationen. Organisation X, Sony Mobile och CGI hade policys och riktlinjer skapade av organisationens juridiska avdelningar som avgjorde om ett OSS fick användas eller ej. Inom både Sony Mobile och Organisation X var alla OSS tvungna att skickas till denna juridiska avdelning för bedömning innan det kunde användas. CGI svarade att de har en tydlig policy över vilka licenser man får eller inte får använda sig av vid utveckling av mjukvara och sedan är det upp till utvecklarna själva att följa denna policy. Det mer centraliserade licensarbetet som dessa organisationer utför går hand i hand med det som enligt litteraturen sägs vara en av de viktigaste förebyggande åtgärderna för att undvika licensbrott (Back & Silic, 2017; Brown & Giera, 2004; Conradi et al. 2010).

LDC var inte säkra att de följde alla licenser de använde sig av, men förklarade detta med att de inte vidareutvecklar någon kod och således besitter en låg risk att bryta mot licenser. På grund av hur universitets licensmodeller fungerar sade LDC att det snarare är större risk att de bryter mot proprietära licenser. Trots att det enligt litteraturen är en kritisk risk att inte undersöka och kontrollera sin licenser (Back & Silic, 2017; Brown & Giera, 2004; Conradi et al. 2010) har LDC arbetat problemfritt med OSS.

5.2 IT Risk Management


5.2.1 Riskstyrning

Den aspekten av riskstyrning som berörts i den här uppsatsen är hur väl ledningen är medvetna och involverade i OSS-relaterade risker. Både Sony Mobile, CGI, och Organisation X hade OSS-policies framtaga av ledningen vilket är en nyckelfaktor vid användningen av OSS enligt Belle et al. (2006). Dock påpekade Sony Mobile att ledningen inte alltid ställt sig positiv till och engagerat sig i OSS och LDC menade att detta existerar än idag. Skillnaden mellan ledningen hos Sony Mobile och LDC är att Sony Mobile:s ledning först var skeptiska

5.2.2 Riskstyrning


6 Slutsats

Vårt huvudsakliga syfte med den här uppsatsen var att undersöka om teorin kring ITRM korrelerar med organisationers hantering av OSS-relaterade risker. Därav var forskningsfrågan följande: *Vilka aspekter av IT Risk Management beaktas när organisationer använder sig av Open Source Software?*


Inom riskstyrning beaktade tre av fem organisationer att ledningen måste vara införstådd i såväl möjligheter som risker som kan medfölja vid användandet av OSS. Här fick vi ofta svaret att det har blivit bättre med tiden vilket vi tolkar som en ökande mognad av OSS inom organisationer.


Utöver användningen av OSS-riskverktyg fanns även manuella processer på plats hos majoriteten av organisationerna för att komplettera analys av OSS-risker som automatiserade verktyg inte kan fånga upp. Här belystes aspekter som community, support, och risk för övergivna projekt.

Riskhantering beaktades unikt inom varje organisation och tog form genom bland annat tydliga policys för hantering av OSS eller genom att enbart använda välkända OSS och således förlita sig på att andra organisationer redan genomfört riskbedömning på mjukvaran. Vi tycker att den sistnämnda aspekten av riskhantering är godkänd då dessa communities och OSS är mycket etablerade och av god kvalitet.

Den aspekt som lyste med sin frånvaro i datainsamlingen var användningen av riskformeln (T * V * A = R). Kontentan var att den antingen var för generell eller för teoretisk vilket vi, och delvis litteraturen, håller med om. Revideringar av formeln har gjorts, både inom litteraturen och av de intervjuade organisationerna. Detta tolkar vi som ett bevis på att det är svårt att beräkna och estimera risker.

De problem och risker med OSS som introducerades i kapitel ett och som utvecklades i kapitel två var överlag inte något som de intervjuade organisationerna kände igen sig i. Detta tolkar vi som att riskhanteringen och riskmedvetenheten är fungerande samt en ökande mognad av OSS inom organisationerna.. Vi är medvetna om att det i kapitel ett presenterades
rapporter och exempel på att riskerna och problemen med OSS existerar och blir till verklighet. Att vår slutsats delvis motsäger detta tror vi beror på att de organisationer som blivit utsatta är mer attraktiva måltavlor än de organisationer vi intervjuat i denna uppsats.

När vi reflekterar över uppsatsen är vi nöjda med den data vi samlat in och tycker att vi har fått ett godtyckligt svar på vår forskningsfråga. Trots att studien varit av kvalitativ ansats där intervjunallen innehöll många öppna frågor har den insamlade datan i hög grad korrelerat med litteraturen som ligger till grund för ämnesområdet. Detta tolkar vi som att både den litteratur vi har analyserat samt de organisationer vi haft kontakt med har varit av mycket god relevans för att uppnå uppsatsens syfte.
7 Referenser


IDATE (2016). DigiWorld Yearbook 2016, Tillgänglig via: [Hämtad 28 Mars 2018]


Koskelainen, A. (2018). Dataintrånget mot Equifax kan ha blivit det dyreste i företagshistorien, ComputerSweden, 5 Mars, tillgänglig via: [Hämtad 2 April 2018]


Linux Foundation (2018). Participating in Open Source Communities, Tillgänglig via: [Hämtad 30 Mars 2018]


W3Techs (2018-b). Usage of content management systems for websites, Tillgänglig via: https://w3techs.com/technologies/overview/content_management/all [Hämtad 22 Mars 2018]


8 Appendix

8.1 Intervjumall

Intervjuperson:
Titel:
Företag:
Antal anställda:
Kort beskrivning av företaget:

Våra begreppsdefinitioner:
Open Source
- Källkod som är fullt tillgänglig för andra att modifiera och distribuera utan begränsningar

IT Risk Management
- Hanteringen av IT-relaterade risker och problem
- Inom OS: säkerhetshål, buggar, licenser, support, dokumentation

Frågor:
- Vad är din uppfattning av begreppet OS-mjukvara?
  o Om annorlunda än vår, presentera vår uppfattning

- Vad är din uppfattning om/av begreppet IT Risk Management?
  o Om annorlunda än vår, presentera vår uppfattning

- Kan ni ge några exempel på OS-mjukvaror ni använder?

- Hur går det till när ni väljer OS-mjukvara?
  o Är det fritt fram för utvecklarna att välja vad de behöver eller har ni specifika riktlinjer?
  o Gör ni någon form av riskidentifiering, riskbedömning, eller riskhantering av OS-mjukvaran?
    • Hur identifieras risker?
    • Hur går riskbedömningen till?
• Uppdateras riskbedömningen kontinuerligt efter att OS-mjukvaran är implementerad?
  • Hur hanteras riskerna?
  • Använder ni något OSS-specifikt riskhanteringsverktyg, som RISCOSS t.ex.?

• Har ni upplevt några problem med OS-mjukvaran? (buggar, support/dokumentation, säkerhetshål)
  o Om ja, vad och hur har ni hanterat de problemen?
  o Om inte, vad hade ni gjort ifall något problem skulle uppstå?

• Hur kontrollerar ni att OS-licenserna följs korrekt?
  o Om nej, hur vet ni att ni inte riskerar legala följder?

• Använder ni något IT Risk Management-ramverk som ISO 27005 eller ISACA:s IT Risk Framework (för att nämna några)?
  o Om ja, hur appliceras det på valet och användningen av Open Source-mjukvara?
  o Använder ni någon formel för att beräkna IT-risker? (R = T * V * A)
  o Har ni anställda som enbart jobbar med detta eller är det något som alla anställda tar del av som en del av arbetet?
  o Hur väl informerade/medvetna är ledningen/”högre instanser” om riskerna med OS?

8.2 Mailmall

Hej,


Vi skulle vilja komma i kontakt med någon som har ansvar eller kunskap om IT-arkitekturen i er organisation. Gärna någon som har koll på val eller inköp av mjukvara. En intervju tar max 30 minuter och önskas anonymitet kan vi ordna det. Personer med följande eller liknande titlar och roller är av intresse:
8.3 Intervjutranskriberingar

8.3.1 LDC

Denna intervju spelades in men ljudfilen förstördes. Intervjun har återgetts så gott som möjligt via de anteckningar som fördes. Intervjuobjektet har även informerats om detta samt godkänt transkriberingen.

**Intervjuperson:** Magnus Persson (MP)

**Titel:** IT-Arkitekt

**Antal anställda:** 100

**Kort beskrivning av företaget:** LDC är ett centralt organ inom Lunds Universitet som ansvarar för många underliggande IT-tjänster som t.ex. drift och säkerhet.

**Intervjuledare:** Axel Hellström (AH)

**Notarie:** Fredrik Moberg (FM)

AH: Vad har din avdelning för ansvarsområde?


AH: Vad är din roll och dina arbetsuppgifter?

MP: IT säkerhet och inom det är jag IT Arkitekt.

AH: Vad är din uppfattning om/av begreppet Open Source-mjukvara?

MP: Egentligen så skulle jag vilja säga att det är FOSS, men det är inte bara det. Vi kör ju två operativsystem om man kollar i våra servrar, Windows eller Redhat. Redhat är inte i sig OSS men dom säljer tjänster som hanterar OSS. Sen har vi annan OSS som vi inte betalar för då vi anser att communityn är så pass stor att vi inte behöver avtal. Tillsammans med Redhat kan vi inte få ett service-avtal men det ingår i Redhat licensen. För IT säkerhet har vi i princip gått över till att köra elastic stack på nästan allting, log stash elastic search. Idag har den så stor community och det finns så många som jobbar i det att där är vi trygga med att den lever så länge som vi behöver den, då är valet enkelt för att vi inte har pengar att köpa något proprietärt. Ska vi ha den här funktionaliteten så har vi inte råd med proprietära val, vi skulle få mycket sämre lösning om vi skulle betala för den, vilket gör valet enkelt. Vi är så fattiga att vi inte kan betala men vi kör gärna OSS. Problemet ligger att vi måste ha leverantörer att skylla på/peka på om något går snett. Gällande till exempel huvudplattformarna måste vi
därför ha en proprietär leverantör för att kunna peka på vid diverse problem.

**AH:** Hur gör ni om ni inte kan peka på någon?

**MP:** Redhat kan vi ju peka på till exempel, om vi tittar på IT-säkerhetsverktyg så kan vi inte peka på någon, det går inte. Om vi står vid valet mellan att hitta något att betala för och att få den funktionen vi vill ha och inte betala men få funktion utan säkerhet, då är det enkelt för oss. Vi kollar på communityn, produkten, livslängd, patchfrekvens för att försöka avgöra om det är värt det eller inte. Universitetet körde openBSD i 10 år förut och vågade köra universitetets brandväggar för openBSDs trackrecord är så pass bra att vi vågade, om något händer med utvecklingen så får vi leva med den sista versionen tills vi har hittat en tjänst eller mjukvara att byta ut den med.

**AH:** Vad är din uppfattning om/av begreppet IT Risk Management?

**MP:** Ju mer IT du har desto större risk utsätter du dig för, man vet hur illa saker och ting fungerar och hur lätt det är att sänka kärnverksamhet i ett öppet nät som de flesta är.

**AH:** Gör ni någon form av riskbedömning, riskanalys eller utvärdering?

**MP:** Alla system som utvecklas, nu pratar vi inte endast OS, utan alla system som vi köper eller utvecklar, allt från stora som ladok ned till småsystem för analys eller liknande, måste hanteras enligt PM3. PM3 är en modell för hur man utvecklar projekt och förvaltar projekt, mer om förvaltning egentligen, och förvaltning av system, systemförvaltare, systemägare. Det är samma för IT-resurs, alla har olika roller och sedan går man igenom sin mall för hur man ska utveckla den, i PM3 ingår även en riskanalys.

**AH:** Hur utförs riskbedömningen?

**MP:** Man bör göra riskanalys om man använder OSS, man måste tänka “vad händer om min infrastruktur försvinner?”. Det har hänt, inte ofta men det har hänt, till exempel kan utvecklarna forka av och bli ovänner och allt delar upp sig.

**AH:** Hur skiljer sig valet av OSS från valet av Proprietär mjukvara?

**MP:** Du har mer säkerhet om du väljer en proprietär mjukvara. Om IBM tar på sig att dom säljer en produkt har dom en livshanteringscykel, dom måste underhålla i flera år trots säljstopp. Den bestäms ofta långt i förväg så att man kan planera sin systemdrift enligt denna. Mozilla och Apache är för stora för att forka till exempel men mindre OSS kan lätt forka, kolla bara på openBSD.

**AH:** Har ni kollat på eller använder ni er av RISCOSS?

**MP:** Nej, vi använder inte det. Det är ganska få som tar med den i sin riskanalys så vitt jag vet. Vad händer om en kärdel i infrastruktur förändras på ett sätt så jag inte kan använda den
längre? Vi har haft saker och ting vi har behövt förändringar på, vi har skickat förslag till OSS-utvecklarna och dom har inte förstått varför det behöver förändras. Vi kan inte få något att förändras utan bara föreslå förslag och hoppas dom implementerar det, vi kan leverera “business impact report” till dom men vi betalar inget och då kan vi inte kräva någon förändring. Hade IBM fått samma rapport hade dom lyssnat då vi troligtvis är en stor kund hos dom.

AH: Har ni upplevt några problem med OS-mjukvaran? Till exempel säkerhetshål, buggar, support/dokumentation etc?


AH: Kontrollerar ni så att OS-licenserna följs korrekt och i så fall hur?


AH: Hur vet ni att ni inte riskerar legala följder?

AH: Använder ni något IT Risk Management-ramverk som ISO 27005 eller ISACA:s IT Risk Framework (för att nämna några)?

MP: ISO 27005 använder vi, och vi försöker arbeta enligt det så gott det bara går.

AH: Skulle du säga att ni implementerar det steg för steg?

MP: Vi försöker följa det på LDC så gott det går. Utöver det är ITIL en bra plattform för att säker och ting ska göras strukturerat, vem gjorde det, hur och när är det klart etc.bara det gör att vi fyller många krav i andra standarder som tex ISO, då är det lättare att införa ISO27000-tänket och de saker som ska göras. På frågan om vi är helt compliant? Nej, men det finns inget krav att certifiera oss, bara att följa den, och gällande den centrala biten följer vi den men kanske inte genom hela verksamheten. Vissa är väldigt dukta men andra struntar i det helt och hållet, när jag började för 30 år sedan var allt centraliserat och skulle precis decentraliseras. Det har det gjort i 15 år men nu börjar det bli mer centraliserat igen, det är jobbigt att sköta, kostar massa pengar och de ekonomiska ramarna blir snävare och snävare. Även om universitetet har mycket pengar, kan vi inte ta det och stoppa in i daglig drift, dom ska användas till satsningar som kan göra saker bättre, det är ett litet problem för oss.

AH: Använder ni någon formel för att beräkna IT-risker? (Risk = Threat * Vulnerability * Assets)

AH: Hur väl informerade/medvetna är ledningen/"högre instanser" om OSS-risker?


AH: Du pratade tidigare om ITIL, skulle du kunna förklara lite mer kring det?


8.3.2 Sony Mobile

Intervjuperson: Carl-Eric Mols (CEM)
Titel: Head of Open Source
Antal anställda: 7100
Kort beskrivning av företaget: Sony Mobile Communications är ett telekommunikationsföretag som tillverkar mobiltelefoner och tillbehör.
Intervjuledare: Axel Hellström (AH)
Notarie: Fredrik Moberg (FM)

AH: Vi kan ju berätta lite om oss och vår uppsats då, vi går sista terminen på Lunds Universitet, systemvetenskap pluggar vi, och vi skriver uppsats, kandidatuppsats. Den handlar ju om som vi skrev i maildet där, hur företag hanterar risker och problem vid användning och val av Open Source mjukvara. Och vi har ju en ganska positiv inställning till OS då, men vi vet också att det finns risker och problem med OS som till exempel säkerhetshål, buggar, licenser, support, dokumentation, såna saker och vissa är ju inte specifika för OS, det finns ju säkerhetshål i proprietär programvara också såklart, men det är dom grejerna vi kollar på där. Så det vi vill ta reda på är om företag tänker på dom här sakerna över huvud taget, om det finns några utarbetade riktlinjer eller processer för att hantera dom här OS-specifika riskerna då. Så det ena vi kollar på är open source då, det andra vi kollar på är IT risk management kallas det, som är som en-, hanteringen av IT-relaterade risker och problem kan man säga

AH: Vad är din uppfattning av begreppet OS-mjukvara?

affärsmodeller. Första har vi den klassiska extenderade modellen, sånt som Red Hat sysslar med. Det vill säga att de distribuerar open source och tar betalt för distributionen, inte för koden. Och de kan de motivera genom att de tar betalt för verktyg som inte är open source och för annat som är bra att ha.

**AH:** Support till exempel?


**AH:** Vad är din uppfattning om av begreppet IT Risk Management?


AH: Vet du om den processen görs enligt NIST som du nämnde tidigare? Är det software security som jobbar efter NIST-modellen?


AH: Så som ett exempel. Om man skulle ta in en open source-licens, kanske läsa igenom den men missa någonting, inkorperera den och sen blir något fel, vad händer då?


AH: Kan du ge några exempel på OS-mjukvaror ni använder?
**CEM:** Främst Android, men vi har Open Source i allting. Verktyg, till och med Corporate IT är glada i Open Source nu för tiden haha. Big data, cloud och IoT och till och med drönare. Sony Mobile i Japan äger ett litet drönarföretag och tittar man i koden för den drönaren så är det GPL v3 i allthopå. Så det är Open Source överallt.

**AH:** Hur går det till när ni väljer OS-mjukvara?


**AH:** Gör ni någon form av riskidentifiering, riskbedömning, eller riskhantering av OS-mjukvaran?

**CEM:** Det vi gör med Android gör vi nu med allt annat också. För Sony Mobile är upp påväg på en resa. Om tre års tid kommer vi inte bara syssla med mobiler utan det finns mycket annat som vi har börjat jobba med, IoT inte minst.

**AH:** Men hur tar just riskidentifiering, riskbedömning, och riskhantering sig i uttryck när ni väljer OS-mjukvaran?

**CEM:** Jo, det jag tänkte komma till är att de processer och strukturer vi har på plats när det kommer till Android, transplanterar vi till andra verksamheter här i Lund också. IoT-gänget jobbar under samma regler och riktlinjer. Det finns ju just nu någonting som är superhot och som alla springer omkring och är väldigt uppmärksamma på är GDPR. Vi började med det här jobbet för två år sedan. För oss är liksom, vi har allting på plats. Vi har processerna på plats.

AH: Använder ni något verktyg för att identifiera, analysera och hantera OSS-risker? Det finns till exempel ett verktyg som heter RISCOSS.

CEM: Ja, men vi använder oss av ett annat verktyg som heter Black Duck Protex. Det är så inbyggt i verktygskedjan att det sker per automatik. När man laddar upp något till vår main branch så kommer det så småningom scannas av och sen scannas det av varje dygn flera gånger osv. Så det är helt automatiserat. Sen har vi sedan en tid tillbaka börjat kolla på den svenska lokalutmanaren, har ni hört talas om FOSSID?

AH: Nej, tror inte det. Men FOSS vet vi ju vi vad det står för.


AH: Har ni upplevt några problem med OS-mjukvaran? (buggar, support/dokumentation, säkerhetshål)


AH: Du pratade om Black Duck-verktyget förut. Ser det till att Open Source-licenserorna följs?


**AH:** Precis, så man kan ju skydda sin Intellectual Property även med Open Source.


**AH:** Jättegärna


**AH:** Använder ni någon formel för att beräkna IT-risker? (R = T * V * A)


**AH:** Hur väl informerade/medvetna är ledningen/"högre instanser" om riskerna med OS?

har fortfarande kvar mindsetet att jag måste frälsa ledningen. Här hos oss är det ju totalt genomslagen. 2012 - 2013 gick det en kulturell chock igenom bolaget då man bara "Nu tar vi emot Open Source helhjärtat".

8.3.3 CGI

Intervjuperson: Philip Vendil (PV)
Titel: Certificate Service Development Team Lead
Antal anställda: 72 000 globalt
Kort beskrivning av företaget: CGI är ett globalt IT-service konsultbolag som styrs från Kanada.

Intervjuledare: Axel Hellström (AH)
Notarie: Fredrik Moberg (FM)


PV: Mm

AH: Så det är det vi håller på med, men vi kan väl börja med om du kan berätta lite om CGI och din roll, dina arbetsuppgifter.

PV: Ja, om CGI, är väl ett jättestort konsultbolag, kanadensiskt bolag som finns i stora delar av Europa och Nordamerika i varje fall. Och jag jobbar ju som huvudutvecklare på Security-delen och vi tar fram en certifikattjänst heter det, vi tar fram säkerhetslösningar, det finns digitala signaturer och PKIer och så, åt kunder. Jag själv har ju väldigt lång bakgrund inom OS, jag har utvecklar OS-mjukvara hela mitt yrkesliv egentligen, så jag har väldigt mycket erfarenhet av det. Just vår avdelning, vi använder nästan uteslutande OS i dom tjänster vi tar fram, för kunder.

AH: Okej, yes, vi skulle kunna börja med att du kort bara berätta din uppfattning om begreppet OS, vad det innebär för dig, så vi ligger ungefär på samma nivå.
PV: Open source för mig. Det är väl, ja, källkoden är tillgänglig, man kan ladda ned, man kan bidra, utbyta idéer, ja, som utvecklare är det ofta att man kan fixa problem själv som man upptäcker som man ofta, om det är en kommersiell produkt så är man ju ganska läst då när det säger stopp, medan vi försöker ju då hitta lösningar och använda OS för att göra liksom helhetslösningar, så att, ja-

AH: Jamen det är ungefär samma som vi har definierat det, sen då nästa område som vi undersöker är det här då som jag sa, IT risk management, har du hört om det, vad tänker du om det begreppet, vad innebär det för dig?

PV: Risk management är ju som jag tolkar, alltså vi gör ju riskanalyser i våra tjänster när vi tar fram problem och hot sådär, så kategoriserar vi det i olika kategorier och sen så gör vi då en åtgärdsplan hur vi fixar det, så att det är vad det står för för mig, det är så vi jobbar.

AH: Ja men okej, det låter bra, vi kommer nog, jag ska säga det också att jag kör frågorna en och en och ifall det kommer en fråga som du känner att du redan har svarat på kan du referera till det eller att du har redan sagt det eller att det var det jag pratade om innan liksom, så att du är med på det.

PV: Ja!

AH: Då kan vi gå in på om du skulle kunna ge några exempel på Open Source mjukvara som ni använder i era tjänster eller lösningar idag?

PV: Ja, alltså det är ofta dom stora, vi har ju mariaDB, det är ju, vi jobbar ju i Java-världen så det finns väldigt mycket open source där. Så det är ju det, spring framework heter det också, jag försöker komma på allt men vi använder nästan uteslutande OS, det är väldigt lite proprietära lösningar som vi arbetar med över huvud taget. Utan alla bibliotek, all utveckling vi gör då, bygger på OS.

AH: Okej, hur går det till när ni väljer OS mjukvara, vi förstår att det är efter behov, att ni har specifika problem som ni måste lösa, men är det fritt fram för utvecklare att välja vad dom behöver eller har ni, du var inne på det lite, ni har specifika riktlinjer då för hur det går till kanske?

PV: Ja vi har ju liksom på företaget en policy om OS som är framtaget, där är det ju specificerat vilka licenser vi får använda så att det inte blir några juridiska problem för CGI eller CGIs kunder då. Till exempel är licensformen GPL förbjuden, det får vi inte ha över huvud taget, för det skulle innebära att kanske kunders egenutvecklade mjukvara måste släppas som OS och det skulle ju inte vara populärt då kanske. Dom har ju definierat vilka licenser som vi får använda,
**AH:** Och det är dom lite mer öppna, tillåtande licenserna då kanske?

**PV:** Ja, det finns ju Apache, MIT, LGPL, får man ju använda. Och den policyn är mest för juridiskt, ofta är det ju juridiska hänseenden som bestämmer det där, just säkerhetsmässiga hänseenden får man sköta själv då. Där har det ju blivit mycket bättre på senare tid tycker jag, med att få säkerhetsvarningar i bibliotek. Tidigare visste man inte alls, då var det väldigt svårt att hålla reda på för att, det är ju liksom beroenden som av OS-bibliotek så det kan vara hundratals bibliotek så det är svårt att veta om den där längst ner på listan har fått en sårbarhet, men där har det ju börjat komma lite sårbarhetsanalyser vet jag. På GitHub vet jag till exempel att man automatiskt kan slå på att den varnar om man har en beroende som har nån rapporterad sårbarhet. Det går också i själva byggmiljön att ha en plugin som kollar upp sårbarheter i dom beroenden man har, det försöker vi ju införa i vårt byggsystem för att fånga dom aspekterna.

**AH:** För det är mycket Open Source projekt som är beroende av andra projekt, det är det du menar med dependencies?

**PV:** Ja, justé, det är små hjälpbibliotek som i sin tur är beroende av ett bibliotek. Ifall man bara drar in ett bibliotek, då kan man få femtio på köpet och då vet man inte vem som har hand om dom och så.

**AH:** Men då har ni, då sa du att det finns GitHub som kan kontrollera det?

**PV:** Ja, det är ett verktyg och sen finns det andra just i byggmiljön så finns det verktyg som slår upp mot den sårbarhetsdatabasen då-

**AH:** Justé, Open Source Vulnerability Database?

**PV:** Ja, CVE heter den väl?

**AH:** Den har vi kollat lite på ja. Vet du, har ni några verktyg för det som ni kör mot koden, eller i byggmiljön?

**PV:** Ja vi håller på att införa det nu för att kunna ha bättre översikt över vad det är som händer och då är det när man bygger så laddar den ner den här databasen och kollar alla beroenden i alla projekt.

**AH:** Okej, kanon! Du var inne lite förrut på att ni gör riskanalys och det har vi också kollat på i det här området IT Risk Management, där har dom då listat olika steg som man ska göra i den här processen, och det är riskidentifiering, riskbedömning och riskhantering. Om vi börjar med riskidentifiering, hur gör ni för att identifiera risker med er OSS användning?
PV: Ja, vi har ju inget specifikt att mäta just för OSS men det ingår i hela tjänsten vi levererar så vi har haft, och det ska man ha regelbundet varje år, så har vi möte där man sitter i grupp och försöker komma på risker på såna här små post-its. Sen går man igenom dom tillsammans, en och en, och bedömer om det är hög sannolikhet och hur stor impact, påverkan, det skulle ha på en skala mellan ett till fyra och sen multiplicerar man dom. Sexton är då när det har högst prioritet och ett har ju minst prioritet på åtgärdsslistan sen. Och sen har vi regelbundna möten där vi går igenom alla punkter som man skulle ha åtgärdat, där man ser status, så att det jobbas ju med regelbundet.

AH: Okej, då har du nog gått igenom lite riskbedömning och riskhantering där att ni följer upp detta och försöker lösa det. Vi har kollat på ett ramverk som heter RISCOSS, är det något som du har hört om? Det är både ett teoretiskt ramverk och verktyg för att hantera risker gällande Open Source, är det någonting du har-

PV: Nej, det har jag aldrig hört talas om faktiskt.

AH: Okej, har ni upplevt några tydliga problem med OSS-mjukvaran som ni har behövt hantera, typ buggar, säkerhetshål, support, dokumentation, något sånt?

PV: Ja alltså det största problemet som jag har upplevt med OSS-användning det är att man väljer projekt som inte är tillräckligt aktivt, och han har tröttnat på det och så är man beroende av det och projektet är dött. Det är en stor risk, det ser jag som det största problemet vi har haft, då får man ta över och underhålla det projektet själv då. Och det är inte så roligt, så att, det är det största problemet. Annars tycker jag OSS är, ja, vissa projekt kan vara riktigt dåligt dokumenterade också, men då har man oftast tillgång till källkoden så att man kan ha nån hum, för det är inte säkert att en kommersiell produkt är bättre dokumenterad, erfarenhetsmässigt hehe. Och då kan det ju vara väldigt krångligt att få hjälp om man ska göra nån support issue, innan man får prata med någon som faktiskt kan något på det företaget så kan det vara väldigt knackigt.

AH: Oftast med lite mer aktiva communities så kanske man få hjälp på forum och liknande?

PV: Ja, riktigt aktiva projekt har ju nästan något så här slack-rum eller IRC-chatt man kan koppla upp sig på så får man oftast väldigt mycket bättre support än en kommersiell supportkanal, faktiskt. Så att just supportdelen har jag inte upplevt att det har varit så jättestora problem med.

AH: Nej, okej, du var inne på det förut, men hur kontrollerar ni att OSS licenserna följs korrekt, för det finns ju en del?
PV: Ja det är ju en utmaning, men det finns ju verktyg som scannar av det där, vi kollar på ett men det var väldigt dyrt så vi har ju gjort det lite manuellt där vi har som rutin att man kollar upp vilken licens det är innan vi kan använda det. Men det kan ju ha beroenden och sånt där, så det är ju lätt att det kan slinka in något, typ en felaktig licens. Det är en utmaning att ha koll på det.

AH: Ja, vet du om ni använder något slags IT Risk Management ramverk? Det finns t.ex. ISO 27005 eller ISACA IT Risk Framework, är det något du har hört om?

PV: Nej, alltså jag vet ju att CGI globalt jobbar mot ISO men det är inget jag har varit i kontakt med under utveckling eller sådär.

AH: Nej, vi är medvetna om det också att det är på en högre nivå, mer för management. Men okej, i det härområdet IT risk Management finns det en formel för att beräkna IT Risker, som är Threat * Vulnerability * Assets = Risk, är det någonting du har stött på eller använt?

PV: Nej men det liknar ju lite vår riskanalys där vi har hot * påverkan, men vi har inte med priset där, det har vi inte. Det är ju dom första två där som vi bedömer utifrån.

AH: Ja, skulle du säga att, hur väl informerade är er ledning eller högre instanser, är dom medvetna om riskerna med OSS? Dels riskerna och, är dom med på användningen som OSS, ser dom det som någonting bra eftersom ni använder det mycket?

PV: Ja, jo men det skulle jag vilja säga att dom vet om. För det var ju en policy som dom tog fram för ganska många år sedan att dom gick igenom vilka licenser man fick använda inom organisationen och sådär.

AH: Det kom ovanifrån då eller vad man ska säga?

PV: Ja, men sedan tror jag att i alla stora organisationer används väldigt mycket OSS över huvud taget, både IBM och Microsoft och dom släpper ju också väldigt mycket OSS nu för tiden. Så jag tror att det är ganska väl förankrat.

AH: Precis, ja, det var dom frågorna vi hade faktiskt, så om du känner att du vill lägga till någonting så får du jättegärna göra det, all information är bra information för oss liksom. Men annars-

PV: Nej jag vet inte om jag har något mer inom området

AH: Då känner vi oss nöjda där, vi får tacka så mycket för att du ställde upp.
8.3.4 Prevas

**Intervjuperson:** Håkan Erlandsson (HE)
**Titel:** Business Unit Manager
**Antal anställda:** 548

**Kort beskrivning av företaget:** Tekniskt IT-företag som erbjuder lösningar, konsulttjänster och produkter till kunder som utvecklar produkter med ett stort IT-innehåll.

**Intervjuledare:** Axel Hellström (AH)
**Notarie:** Fredrik Moberg (FM)

**AH:** Vi kan ju berätta lite om oss och vår uppsats då, vi går sista terminen på Lunds Universitet, systemvetenskap pluggar vi, och vi skriver uppsats, kandidatuppsats. Den handlar ju om som vi skrev i mailen där, hur företag hanterar risker och problem vid användning och val av Open Source mjukvara. Och vi har ju en ganska positiv inställning till OS då, men vi vet också att det finns risker och problem med OS som till exempel säkerhetshål, buggar, licenser, support, dokumentation, såna saker och vissa är ju inte specifika för OS, det finns ju säkerhetshål i proprietär programvara också såklart, men det är dom grejerna vi kollar på där. Så det vi vill ta reda på är om företag tänker på dom här sakerna över huvud taget, om det finns några utarbetade riktlinjer eller processer för att hantera dom här OS-specifika riskerna då. Så det ena vi kollar på är open source då, det andra vi kollar på är IT risk management kallas det, som är som en-, hanteringen av IT-relaterade risker och problem kan man säga

**AH:** Kan du kort berätta om Prevas, vad ni gör samt din roll och lite så?

**HE:** Ja, lite snabbt, Prevas är ett teknikkonsultföretag och vi jobbar framförallt mot företags utvecklingsavdelningar men också mot deras produktionssajter. Vi jobbar med i princip alla ingenjörsdisciplinerna från mekanik, elektronik, embedded programmering, lite högre nivå av programmering, kvalitetssäkring lite kort. Och så automation inom programmering inom IT-sidan då. Jag är ansvarig för den biten som heter PDS det vill säga vi som jobbar mot utvecklingsavdelningarna, så har jag en kollega som ansvarar för dom som jobbar mot produktionen. Vilket innebär att jag har mekanik, elektronik, embedded och applikationsmjukvara i princip, plus QA-biten, under mina vingar så att säga. Och där erbjuder våra tjänster till våra kunder, både att vi kan ta helhetsprojekt-åtaganden och utveckla en produkt åt folk, men också ”vanlig konsulting” då, att vi tillsätter resurser till kundens projekt helt enkelt. Vi jobbar hos dom, sitter hos kunder. Om vi går in på biten med OS som jag misstänker är det som är mest intressant för er-

**AH:** Ja precis vi skulle kunna börja med att vi tar din uppfattning av begreppet av OS så att vi ligger på ungefär samma plan?

**HE:** Mm, jag har själv en bakgrund som civilingenjör, jobbat ungefär 10 år som utvecklare, projektledare, systemingenjör sen, ja 15 år till och med kanske, och sen lite mer åt konsultchefs-hållet. Så dom senaste åren har jag inte programmerat själv eller varit utvecklare eller konstruktör så att säga. Men jag har träffat en hel del kunder och varit inblandad i en hel
del projekt, framförallt på medicintekniksidan. Där var det som så att OS var kanske inte så populärt på medicintekniksidan, för där finns en hel del regelverk som innebär att man kan få det ganska tungt att validera dom här stora mjukvarupaketen. Å andra sidan har man mognat lite grann i den insikten och framförallt efter att många i USA har insett att den här OS-mjukvaran är faktiskt inte så dålig kvalité eftersom det är väldigt många som har tittat på den koden. Och man har då kunnat som medtech-företag faktiskt kunnat ta in koden och sen validera dom bitarna som man använder och göra en riskbedömning på vad kan gå fel här, istället för att kanske behöva gå igenom hela koden rad för rad, vilket i praktiken är omöjligt. Och när man har gjort det så, hehe, så har vi ju en ny version som har ändrat sig. Men mina erfarenheter där gällande OS-kod är ju två bitar, för ett tag sedan började dom här bitarna komma in rätt mycket och då uppenbarade sig dom här licensavtalen som kommer med OS, det var rätt många som inte insåg vad det innebar. Där har jag varit med om rätt många projekt där vi går igenom företagens OS, vilka licensavtal har dom faktiskt tagit in i sitt paket som dom i sin tur säljer. Där är det en del företag som har råkat väldigt illa ut, där finns möjligheter att titta på det, bland annat finns det verktyg för det, till exempel ett företag som heter blackduck, som jag misstänker att ni känner till?

AH: Ja, det har vi hört talas om.


AH: Nej, det börjar bli många klassiska företag, börjar ju nu bli mjukvaruföretag också.

HE: Precis och dom är rätt så fokuserade på hur ska vi nu hänge med i det här, hur ska vi kunna tjäna pengar på det och kanske misstänker jag att man, den här riskhanteringen är inte det som ligger högst upp i prioriteringsordningen. Och där tror jag, där har vi nog inte sett vad som kommer hända ännu, utan hela problemet är på väg att byggas upp nu.

AH: Precis, och det är det andra vi undersöker då, det här med Risk Management, IT Risk Management har vi kollat på vilket är ett begrepp över hur man hanterar risker. Har du hört det, har du kommit i kontakt med det?

AH: Ja det händer ju grejer hela tiden, om vi går tillbaka till OS lite, kan du ge några exempel på OSS ni använder? Ni är ju ute mot kunder mest kanske då, men har du några exempel?


AH: Okej, hur går det till när ni väljer OS, om ni jobbar mot en kund och väljer OS, är det upp till- är det fritt fram för dom konsulterna som jobbar med det att välja eller har ni några specifika riktlinjer för det där?

HE: Vi har inga specifika riktlinjer utan det är helt beroende på kunden. Ibland är det så att vi kommer in på projekt och så har kunden redan valt att man ska använda ditten och datten, operativsystem eller OS eller vad det nu kan vara. Då är det bara att göra det, ibland kommer vi in tidigare i projekt där vi faktiskt får göra en förstudie och ta reda på lite grann vilken typ av verktyg ska vi använda, hur ska systemet se ut och ska vi använda OS eller inte. Det har ju mycket av dom här bitarna, så det beror mycket på när var och hur i projektet.

AH: Juste, gör ni någon form av riskanalys eller riskidentifiering, riskhantering av dom här OS-mjukvarorna?

HE: Ja! Det försöker vi göra i den mån vi får lov av kunden, eller får tid eller där vi kan övertyga kunden om att ja men det är nog rätt vetligt att göra det. Eller så har kunden själva gjort det tidigare och tycker att det räcker så. Men vi försöker ta reda på för kunden, är ni medvetna, har ni gjort någon riskanalys, är ni medvetna om vad som händer, har ni vägt in dom parametrarna i valet?

AH: Har ni någon metod, något ramverk eller verktyg som du pratade om tidigare, Black Duck då, som ni använder för att analysera OS-mjukvara?
HE: Jag ska säga att jag vet faktiskt inte, jag är lite för ny inom Prevas, jag har bara varit här i lite drygt ett halvår så att, det kan jag inte säga.

AH: Nej, vet du om några konkreta exempel på när ni har upplevt problem med OS, till exempel som vi pratade om licenser, dålig support eller dokumentation, något sånt?

HE: Nej, det ska jag inte säga att jag på rak arm kan, tvärtom är det väl lite så att OS har, väljer man en hyfsat vanlig är det rätt stora communities där man kan gå in och ofta får man väldigt bra svar rätt snabbt. Så supporten där är nog rätt så bra faktiskt.

AH: Det brukar ju vara en säkerhet i sig där att om ett OS-projekt är stort och mycket använt så är det oftast bra.

HE: Precis.

AH: Det här med OS-licenser, du berättade om någon gång ni kontrollerade att OS-licenserna var korrekt. Skulle du kunna utveckla lite mer kring hur ni gjorde om du kommer ihåg?

HE: Det var faktiskt några år sedan nu men vi, vad vi gjorde var egentligen en gap-analys på vilka OS-paket hade man och hur hade man behandlat deras licenser.

AH: En gap-analys?

HE: Alltså, var är man och var borde man vara, hur mycket är det kvar att göra för att faktiskt uppfylla dom licenserna som man mer eller mindre medvetet hade plockat in i sin produkt. Det var nog så att det blev en liten sån aha-upplevelse för kunden att, okej var det så mycket att göra det hade vi inte riktigt räknat med.

AH: Nej, just det kan ju lätt bli en licensdjungel där. Vi har ju kollat lite på dom här IT Risk Management-ramverk eller metoder, det finns ju något som heter ISO 27005 som är en sån industristandard, ISACAs IT Risk Framework, är det något du har hört om eller är det något ni jobbar med?

HE: Nej det ska jag erkänna att jag känner inte igen just dom siffrorna av ISO standarden.

AH: Okej, sen i dom här ramverken finns det några, en formel för att beräkna IT-risker, den är väldigt generell och den är Threat * Vulnerability * Asset = Risk.
HE: Precis, den funkar ungefär som dom flesta riskanalyser gör

AH: Nej exakt den är ju inte speciell för just IT, men den känner du igen?

HE: Japp

AH: Har ni använt den någon gång?

HE: Jag kan inte riktigt säga om Prevas har använt den, jag tror det, men jag har inget direkt jag kan säga.

AH: Okej, har du märkt av något? En del av IT Risk Management är ju också det här att man, det är viktigt att ha ledningens stöd, att ledningen är medveten om olika IT-risker som finns. Är det något du har märkt, att ledningen inte informeras tillräckligt om risker som finns med IT, eller möjligheter som finns med IT? Har du märkt av friktionen där eller att det inte riktigt förmedlas?

HE: Ja, det är nog ganska vanligt faktiskt. Och det är lite beroende på vilken typ av kund, bransch men också vad ledningen har för bakgrund, i företaget så att säga. Har dom teknik- och IT-medvetande eller inte, det gör en jättetröskel skillnad ofta. Vad ledningen själva har för preferenser och erfarenheter. Där har jag sett allt från extremt kompetenta ledningsgrupper i företag till, om inte totalt ignoranta eller okunniga så på gränsen till vad man säga.

AH: Okej, då var det nog bra för vår del om du inte har något mer att tillägga?

HE: Nej!

8.3.5 Organisation X

Intervjuperson: Person X
Titel: Head of Vendor Management and Governance
Företag: Organisation X
Antal anställda: 330
Kort beskrivning av företag: Bank

Intervjuledare: Axel Hellström (AH)
Notarie: Fredrik Moberg (FM)

AH: Vi kan ju berätta lite om oss och vår uppsats då, vi går sista terminen på Lunds Universitet, systemvetenskap pluggar vi, och vi skriver uppsats, kandidatuppsats. Den handlar ju om som vi skrev i maillet där, hur företag hanterar risker och problem vid användning och val av Open Source mjukvara. Och vi har ju en ganska positiv inställning till OS då, men vi
vet också att det finns risker och problem med OS som till exempel säkerhetshål, buggar, licenser, support, dokumentation, såna saker och vissa är ju inte specifika för OS, det finns ju säkerhetshål i proprietär programvara också säkert, men det är dom grejerna vi kollar på därt. Så det vi vill ta reda på är om företag tänker på dom här sakerna över huvud taget, om det finns några utarbetade riktlinjer eller processer för att hantera dom här OS-specifika riskerna då. Så det ena vi kollar på är open source då, det andra vi kollar på är IT risk management kallas det, som är som en-, hanteringen av IT-relaterade risker och problem kan man säga

AH: Men du kan väl berätta lite kort om dig själv och din roll och så?


FM: Jag har suttit och spärrat kort hos Organisation X på nätterna via ett annat företag, så jag vet lite grann men inte jättemycket!


AH: Okej
X: Genomsnittsåldern på dom utvecklarna är väl 72 snart hehe, det är ju bara dom som håller på med det

AH: Haha, ja det några få som har den kunskapen fortfarande. Men vad bra, vi skulle kunna gå vidare och fråga vad din uppfattning om begreppet OSS är, så att vi ligger ungefär på samma plan?


AH: Ja just, vi kommer lite till det sen också. Det ena vi undersöker är då OSS, det andra är det som kallas IT Risk Management, vad är din uppfattning om begreppet, har du hört det innan?

X: Om IT Risk Management?

AH: Ja, precis.


AH: Juste, bra! Om vi kommer tillbaka till OSS då, har du några exempel på OSS ni använder just nu eller har använt?


X: Haha, okej, ja eller så säger jag någonting helt annat!

AH: Haha, ja precis då blir det svårt. Men, du har varit inne lite på det, hur går det till när ni väljer OSS? Då förstår jag att det inte är fritt fram för utvecklarna att välja precis vad dom vill ha utan dom måste genomgå någon slags process för detta då?

X: Ja precis det är lite beroende på hur, i vilket skede som det är. Om vi säger att vi ska etablera en helt ny funktion, någon funktion som idag inte finns och där OS kan bli aktuellt, då har vi normalt sett ett projekt som etablerar det här och då sker det urvalet i projektet under överseende av projektledaren men också en blandning av lösningsarkitekter som då blir involverade beroende på vilken teknikstack det handlar om. Vi tar ganska mycket stöd när det kommer till detaljerade tekniska frågor så har vi ganska mycket stöd från koncernen där vi har en central funktion med mer tekniktunga arkitekter. Våra är mer verksamhetsvända. Sker det inom en befintlig funktion där man gör en mindre förändring och det avser skifta ut någonting mot OS eller ändra någonting, då har vi det systemets tekniska ägare som gör urvalet och föreslår att "jag skulle vilja göra den här OS-komponenten" det slutar ändå med samma process för att godkännandet ska gå igenom där arkitekterna och säkerhetsfunktionen ska godkänna komponenten och det handlar väl också om att vi också kontinuerligt följer upp det. Något som är lite svårt, det är ännu en process vi använder oss av som kallas Risk och Regelverksanalys som syftar till att alla förändringar som sker någonstans inom bankens miljöer ska gå genom en sån här analys. Där blir det jätte svårt då man tar in någonting som ständigt förändras, då vet man inte när man ska göra analysen, där vi får vi fortfarande ta ställning till när förändringen är tillräckligt stor för att faktiskt lyftas upp. Om vi kan göra en bedömning att den här påverkar vår riskexponering på något sätt, då måste vi göra en sån här Risk och Regelverksanalys. Utomsten av den kan ju antingen bli att

AH: Då förstår jag att ni gör det här, ett godkännande då, vet du något mer ingående hur den processen går till? Vi har till exempel identifierat i det här IT Risk Management som du har varit inne på också, så finns det tre stora delar, identifiering, bedömning och hantering. Vet du om dom på något sätt följer den mallen eller den metoden eller hur det går till?

X: Ja jag skulle vilja säga godkännandet är lite vid sidan av, vår riskprocess den löper ju på allt vi gör, oavsett om det är OS eller inte. Risker flödar genom organisationen, vi gör riskutvärderingar, med jämna mellanrum och det finns många inflöden till det. Utöver den sker också ett godkännande, normalt sett är det den normala riskrutinen som säger att riskägaren är samma person som ansvarar för systemet, eller för den delen för affärsprocessen som flödar genom systemet. Då blir det ju lite så här att man ställer krav på sig själv. Medan godkännandeprocessen för OS, den är ju, den görs av samma funktioner det är arkitektur och säkerhet alldeles oavsett vilken del av banken som avser implementera den här OS-funktionen.

AH: Okej

X: Dom överlappar/kompletterar varandra lite beroende på hur man väljer att se på det.

AH: Ja, det finns en rätt generell riskprocess där.

X: Mm, sen i vanlig riskutvärdering, det har vi också en mall för som jag skulle kunna skicka över som underlag, ganska simpelt där man gör en matris med risk, konsekvens och så har vi graderingar i fem steg tror jag att det är. Där plottar man ut risken och över en viss nivå blandas flera (?) (samtalet bröts lite grann), lägst upp i högra hörnet ja då är det i princip VDn som måste besluta om vi kan leva med den här risken eller inte och nere i vänstra hörnet då är den så liten att då kan en delegerad del av organisationen ansvara för risken själva.

AH: Juste, vi har ju kollat lite på liknande, vad ska man säga ramverk som är mer specifika för OS, det finns ett som heter RISCOSS och OSS Maturity Model, är detta någonting du har hört om eller någonting du vet om?

X: Nej, faktiskt inte, jag tror att våran OS-godkännande, den är nog väldigt intern. Den är nog i sin absolut simplaste form, det är i stort sett upptäck, review och godkänn.
AH: Okej, kanon. Vet du om att ni har upplevt några problem med just OS som buggar, för lite support/dokumentation, säkerhetsbrister?

X: Nej, sen ska jag säga att vårt användande är ganska begränsat och mig veterligen har vi nog inte tagit in det för någon funktion som är superduper-verksamhetskritisk. I den mån som vi har haft problem så har det inte fått så stora konsekvenser. Sen är jag ganska säker på att det kommer hända så vi utnyttjar mycket mer publika funktioner och standardiserade funktioner, så att det kanske kommer uppstå.

AH: Ja, det kommer säker med användningen. Vi var inne lite på compliance och sånt förut. Det finns ju OS-licenser också, OSS är licensierad på olika sätt, vet du hur ni kontrollerar att dom följs korrekt och att det inte bryts mot där och så?

X: Ja, vi har inom koncernen en central hantering av Software Asset Management och licenser därtill som, jag ska väl inte säga att vi slaviskt följer den men dom allra flesta inköp av licenser görs där genom. Då köper vi kontrollen av det som tjänst så att säga, både hjälper till att förhandla fram i den mån det är förhandlingsbart gällande licensvillkor men också ser till att vi, det finns lika många licensmodeller som det finns licenser näst intill så att det är en hel vetenskap att hålla koll på det. Där vi försöker i god mån nytta den centrala funktionen, och då kan vi dessutom göra inköpen gemensamt mellan bolag i koncernen, det är inte ovanligt att någon annan har samma behov någon annanstans inom en stor koncern.

AH: Juste

X: Exakt hur dom jobbar med det vågar jag faktiskt inte, jag kan förmedla en kontakt till det men jag vet faktiskt inte hur dom, om dom har något verktyg eller vad dom håller på med.

AH: Nej precis, vi vet ju om att det berör många olika delar som kan vara svårt för en person att hålla koll på, men det är bra. Använd ni något IT Risk Management ramverk eller standarder som t.ex. ISO 27005 eller ISACAs IT Risk Framework?

X: Vi är väl inte ISO 27005-certifierade men det mesta av det vi gör är ju inspirerade av, det är ju inte svar på er fråga men vi har ett, ERC heter det, det är någon hyfsat stor och känd programvara för riskhantering som ju de facto, eftersom den är stor och hyfsat standardiserad så styr den in oss mot best practice. Men jag vet faktiskt inte om vårt ramverk är kopierat från nåt av dom eller om det är vårt egenuppfunna men jag skulle tro på det förstnämnda.

AH: Att det är kopierat?

X: Ja att det är åtminstone väldigt hårt inspirerat från något större ramverk.
AH: Precis, många av dom ramverken är ju väldigt snarlika när vi har kollat på dom och jag tänka mig att den processen som finns i dom ramverken är det man ofta kör efter i organisationer bara att man kanske inte har exakt, att man säger att man implementerar ett ramverk till punkt och pricka.

X: Nej exakt, jag vill ju tro att det är någon auditor/konsult som har implementerat det på vårt företag och då gör dom det enligt sin best practice, sen att vi kallar det något annat.

AH: Ja precis, okej, något annat som dom här ramverken har gemensamt är en formel för att beräkna IT-risker som är, eller risker överlag, som är $T \times V \times A = R$, det är en sån som vi har stött på ganska mycket nu, är det något du har hört om eller vet du om ni jobbar med det här?

X: Ja, vi jobbar med något annat som jag tror hör till finansbranschen som är någon förkortning som hette, IKLU hette den nog, Intern Kapital något annat. Där man (samtala bremsa lite) där inträffade risker, man tog med också verkliga inträffade incidenter som ett mått på de hår riskerna. Sen var man tvungen, vi är ju åtagda att ha kapitalräkning baserat på vilken, ja om man kommer till en hög siffra så behöver vi ju mycket pengar på vårat sparkonto för att hantera dom riskerna som om dom väl slår in. Om det blir konsekvenser eller när dom blir det på riktigt, så sånar revisioner gör vi en gång per år för att säkerställa att vi har kapitalräkning för dom riskerna som vi utsätter oss för utefter vår riskaptit. Jag tror och är rätt säker på att det där är något i finansbranschen känt sätt att göra det på, jag tror att dom kontrollerar dom riskerna på det sätt.

AH: Okej, justa, något annat som dom här ramverken nämner väldigt, eller, trycker på är att ledningen eller högre instanser inom organisationen ska vara medvetna om riskerna med IT, i vårt fall med OS, hur känner du dig att det arbetet fungerar hos er, är ledningen med på det så att säga eller är det att det sköts på lägre nivå?

X: Ja men det tycker jag att vi är ganska duktiga på, vår senaste VD, eller hen som sitter nu, har kom in i ett läge där det var ganska lösa (volym?) och fick det i tydligt uppdrag från styrelsen att få kontroll på bland annat det här. Vi har ju en väl utvecklad eskaleringsväg för risker, varenda avdelning på banke sitter kvartalsvis med en kommitté där man får rapportera av på sin riskkarta som den ser ut just nu och eskalera vid behov eller för den delen berätta om man har kontroll på sina risker. Eftersom min chef som är CIO på banke har den här rapporteringssägen till sin chef så behöver ju han ha kontroll på sina detaljer ganska mycket. Vi har ju våran motsvarighet till det då inom IT-organisationen varann vecka där våra risk managers stämmer av med CIOn vad progressen är på alla åtgärder för dom risker vi har identifiserat. Det är en ganska levande organism våran riskkarta som vi ständig jobbar med.

AH: Det finns en kommunikation där då. Då tror jag faktiskt att vi är klara där!
FM: Vi kommer ju transkribera det här så fort vi har tid och så kommer vi skicka det till dig så gott vi har kunnat sammanställa det så får du bara godkänna det innan vi använder oss av datan såklart.

X: Absolut

AH: Om inte du har något att tillägga så är vi nöjda så faktiskt.

X: Nej, inte vad jag kan komma på nu!