Prüfungsprotokoll Vertiefung - Freiling geb. Gärtner Verlässliche verteilte Systeme I+II (je V4), Information Warfare (V2), Fallstudie von Apollo bis Space Shuttle(V2)

14. Juli 2005

Beisitzer war Maximillian Dornseif. Note: 1.0.

1 VVS II

- Mit welchem Thema Anfangen? *(CHT komplett erklärt, was zeigt Beweis, wozu, wie)*
- Das ist ja recht kompliziert, gibt es auch Abstraktionen bei denen das einfacher geht, und warum? *Bsp. TRB (P) oder NBAC (?P)*
- Warum könnte das dann bei Consensus so kompliziert sein *ich überlege mir, weil bei Consensus keine Information mehr besteht, von welchem Prozess was kommt, sage diese Information ist im CHT-Beweis dann im DAG (im Nachhinein: das ist natrlich bei NBAC auch nicht anders...)*
- *will darauf hinaus, dass fr NBAC oder TRB explizit Fehlerverhalten in der der Spezifikation erwähnt wird, bei Consensus nicht. Dies hat auch einen besonderen Namen, den er meinte auch mal in der Vorlesung erwähnt zu haben. Ich kannte das nicht, aber es machte auch nichts. Überhaupt erzählte er jetzt sehr viel, was ber die Vorlesung herausging, wobei ich aber immer wieder auch Anmerkungen gemacht habe, somit wurde dies ein richtiges Gespräch. Es ging auch um den schwächsten Fehlerdetektor für NBAC
- *Auf meine Erwähnung hin, dass das Problem bei TRB einfach ist, da ja P der stärkste Fehlerdetektor ist*: Eigentlich gibt es auch noch stärkere Fehlerdetektoren als P, ist denn vollständige Synchronität äquivalent zu P? *ich kann durch P zwar Runden erzeugen, aber nicht die Dauer begrenzen*
- ja..., *er zeigt mir dann, dass es noch ein stärkerer Fehlerdetektor ist, wenn man erkennen kann, ob vor einem Absturz etwas gesendet wurde, dies allerdings auch etwas ber unser Framework hinausgeht, da man auch die Kanäle miteinbezieht*
- Was ist denn der schwächste Fehlerdetektor für BA? *also mit Muteness-Detektor haben wir es implementieren können, aber vieleicht gibt es ja noch einen schwächeren*
- Die Frage ist auch etwas unfair, denn so genau haben wir Fehlerdetektoren hier garnicht definiert. *Genau, wir hatten auch ja garkeine zum vergleichen. Aber Mutness ist schon ziemlich stark, viel mehr kann man ja bei byzentine-Prozessen nicht verlangen.*

- Und Mutness ist ja sehr protokollabhängig *Ja, verdächtigen bedeutet ja nicht, dass der zugehörige Prozess byzantinisch ist.*

- Wie lassen sich unsere byzantinischen Protokolle auf Security bertragen? *Einmal ist das Problem der Spezifikation. Integrität und Verfgbarkeit lässt sich mit Safety und Lifeness spezifizieren, nicht aber Vertraulichkeit, hierfr hatten wir Non-Interference. Das andere ist, fr unsere Algorithmen hatten wir immer die Annahme, das weniger als ein Drittel der Prozesse byzantinisch wird. Diese Annahme ist fr Security-Probleme nicht mehr realistisch*

2 VVS I

- Wie sinnvoll ist es denn fr die Praxis den schwächsten Fehlerdetektor für ein Problem zu bestimmen? **Hier weiß ich nicht mehr genau, es ging unter anderem um Überdeckung, dass ein Algorithmus für ein assynchrones System Überdeckung 1 hat, eins mit P wird praktisch sehr teuer.**

- Wie funktioniert den Authentifikation bei z.B. einem Linux wie hier? *Passwörter, PAM, z.B. public-key mittels ssh, allgemein ja drei Methoden, wobei Denning (Information Warfare) ja sogar noch eine vierte identifiziert hat...*

- Ach ja, welche denn? *Über den Ort, sie gibt da als Beispiel irgendwie per GPS-Fingerprint, aber ob man das nicht auch unter ëtwas wissenëinordnen könnte?*

- Ja, aber man macht das ja auch so häufig, wenn man in einem Gebäude ist. *Das stimmt, aber eigentlich wurde man dann schon durch etwas anderes authentifiziert*

- Wie ist das denn bei Linux implementiert? *uid, gid, Dateiattribute*

- Und die Passwörter, die werden einfach mit Textstrings verglichen? *Nein, es werden nur die Hashes gespeichert (erklärt, crypt erwähnt)*

- Ist das denn sicher? **über Passwortsicherheit im allgemeinen und herausfinden bei bekannten Hashes im Besonderen philosophiert**

- Für ssh nimmt man ja meist ein anderes Verfahren, was ist daran besser? *es wird nur der public-key auf dem Server gespeichert, daraus kann man keine Informationen ber den private-key gewinnen*

3 Information Warfare

keine extra Frage

4 Fallstudie: Apollo bis Spaceshuttle

- zu Fallstudie: Space Shuttle ist ja aktuell und auch viel in der Kritik, z.B zu teuer, zu gefährlich. Was meinst du, ist die Space Shuttle total veraltet, ist sie sicherer geworden, nach den Unglcken? **Hier ein paar Ideen gebracht, hängt für spätere Prüflinge wohl auch davon ab, wie die nächsten Starts verlaufen**