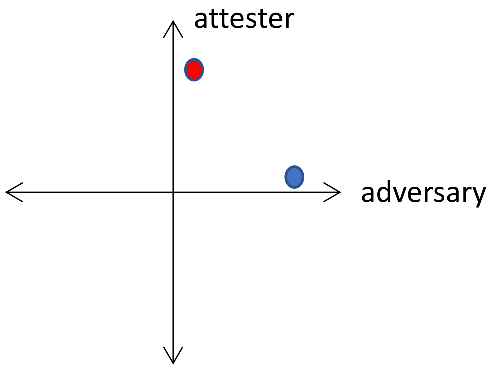
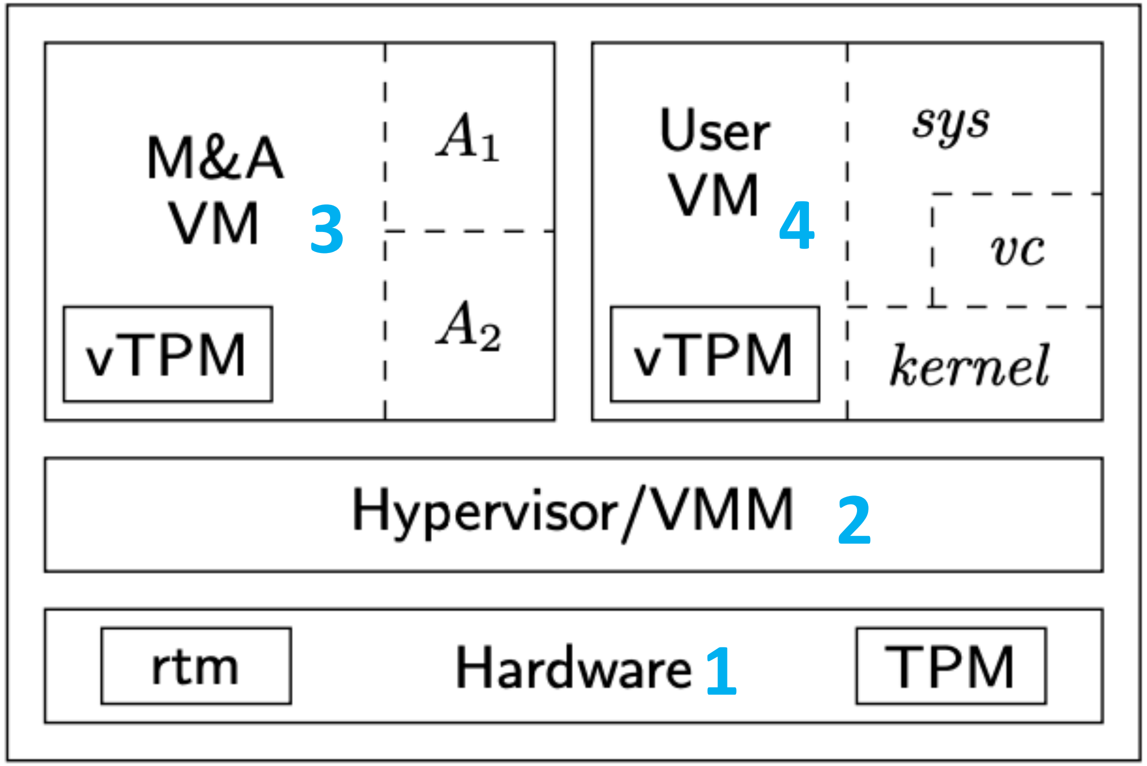
Goals of cost analysis

- Ultimate goal: guide selection of a protocol
- How:
    - Systematic variation of assumption
    - Assign abstract cost to each component that's corrupted
- Consider:
    - Cost to adversary
    - Cost to attester

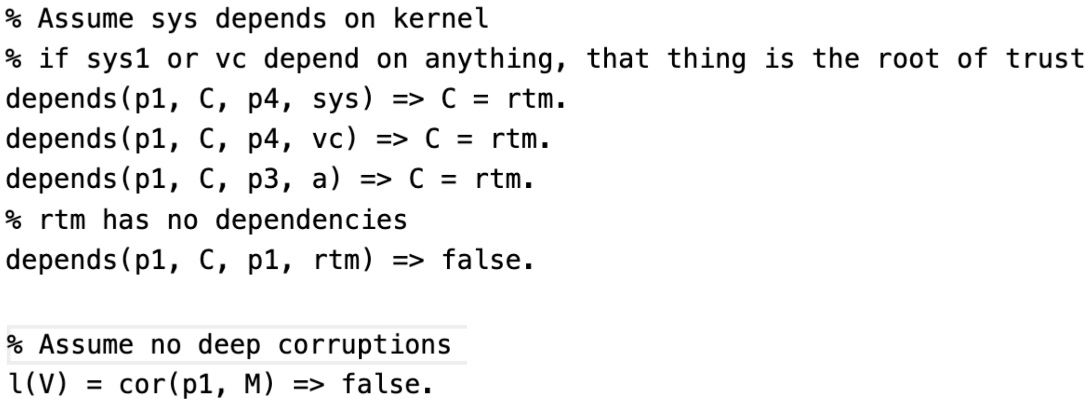


Say we have the architecture from "Confining the Adversary" Paper

- ms(rtm, A1)
- ~~ms(rtm, A2)~~
- ms(A1, vc)
- ~~ms(A2, ker)~~
- msker (vc, sys)



Control Variables



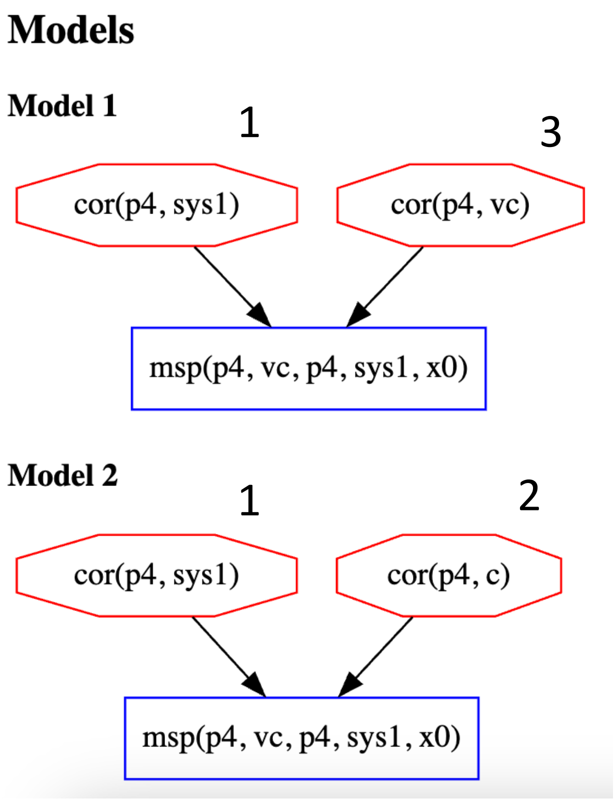
```
% Assume sys depends on kernel
% if sys1 or vc depend on anything, that thing is the root of trust
depends(p1, C, p4, sys) => C = rtm.
depends(p1, C, p4, vc) => C = rtm.
depends(p1, C, p3, a) => C = rtm.
% rtm has no dependencies
depends(p1, C, p1, rtm) => false.

% Assume no deep corruptions
l(V) = cor(p1, M) => false.
```

- Assumptions
  - Always assume recent/deep
  - Make no assumptions about system dependencies except…
    - TPM is the root of trust… has no dependencies
    - Virus checker and system depend on the hardware (p1,rtm)
    - A1 depends on the hardware (p1,rtm)

Side note: I changed all theory files to the original… allows for corruption only at the same place

- If I made it allow for corruption at different places… CHASE seemed to introduce corruption events with odd labels sys

First protocol…. Just measure *sys* using *vc*



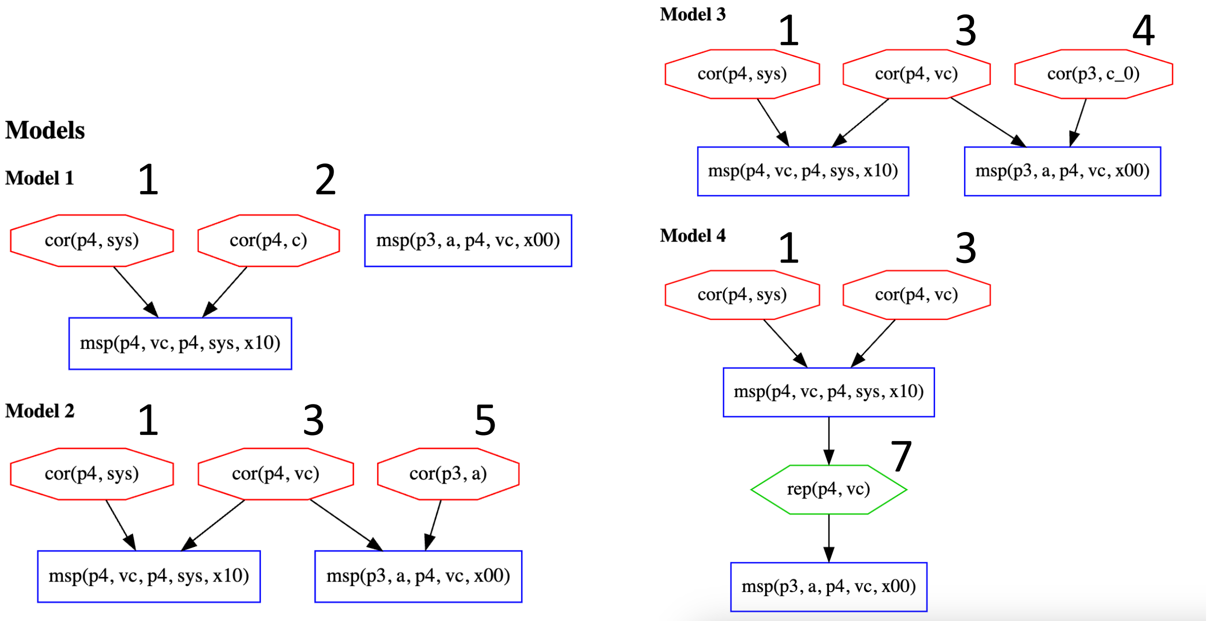
Event	Cost
cor(p4,sys1)	c1
cor(p4,vc)	c3
cor(p4,c)	c2
MODEL 1 COST	c1 + c3
MODEL 2 COST	c1 + c2

If you add in dependencies about a then labels to corruption events change vc-sys-par

Measure *vc* and *sys* in parallel

- Protocol
  - *target: @p3 [a p4 vc]
    +~+ @p4 [vc p4 sys]

```
% Assume dependencies
% if sys1 or vc depend on anything, that thing is the root of trust
depends(p1, C, p4, sys) => C = rtm.
depends(p1, C, p4, vc) => C = rtm.
depends(p1, C, p3, a) => C = rtm.
% rtm has no dependencies
depends(p1, C, p1, rtm) => false.

% Assume no recent corruptions
prec(V, V1) & l(V1) = cor(P,C) & ms_evt(V)
=> false.

% Assume no deep corruptions
l(V) = cor(p1, M) => false.
```



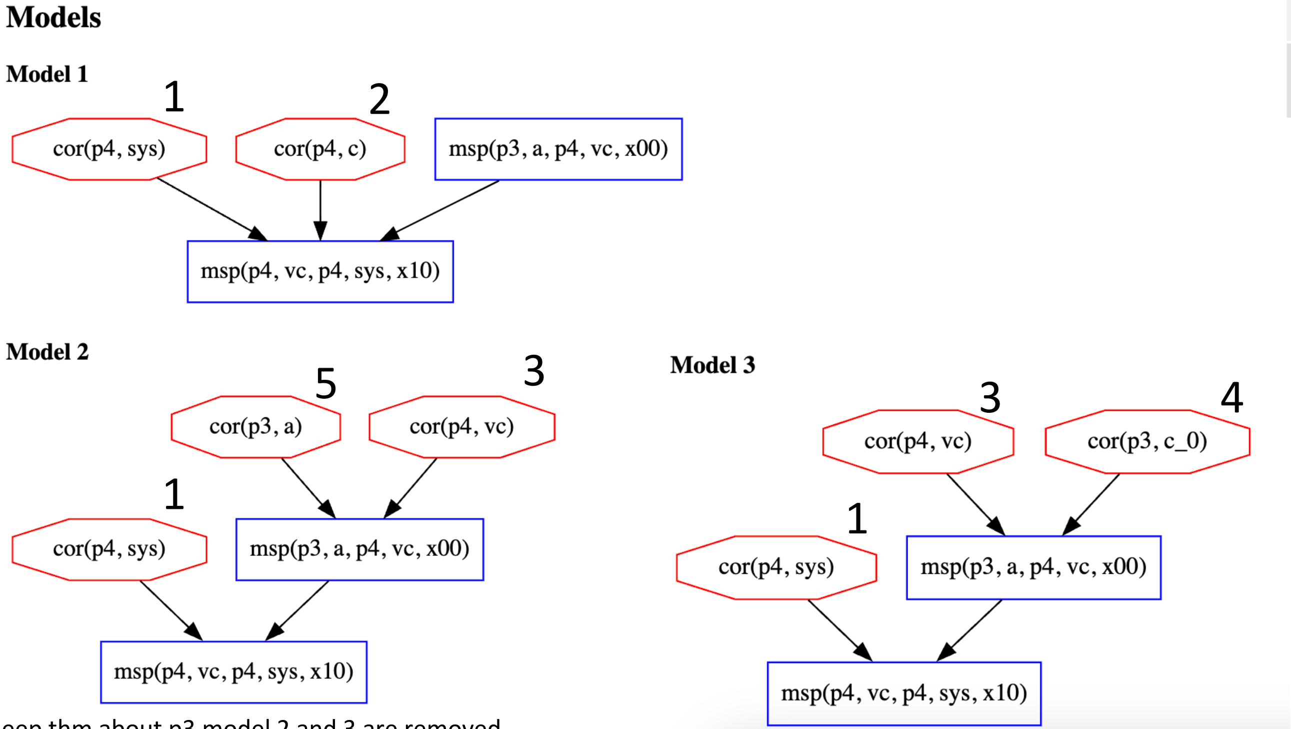
Model	Total cost
1	c1 + c2
2	c1 + c3 + c5
3	c1 + c3 + c4
4	c1 + c3 + c7

vc-sys-seq

Measure *vc* and *sys* in sequence

- Protocol
  - *target: @p3 [a p4 vc]
    +<+ @p4 [vc p4 sys]

Model	Total cost
1	c1 + c2
2	c1 + c3 + c4
3	c1 + c5 + c3

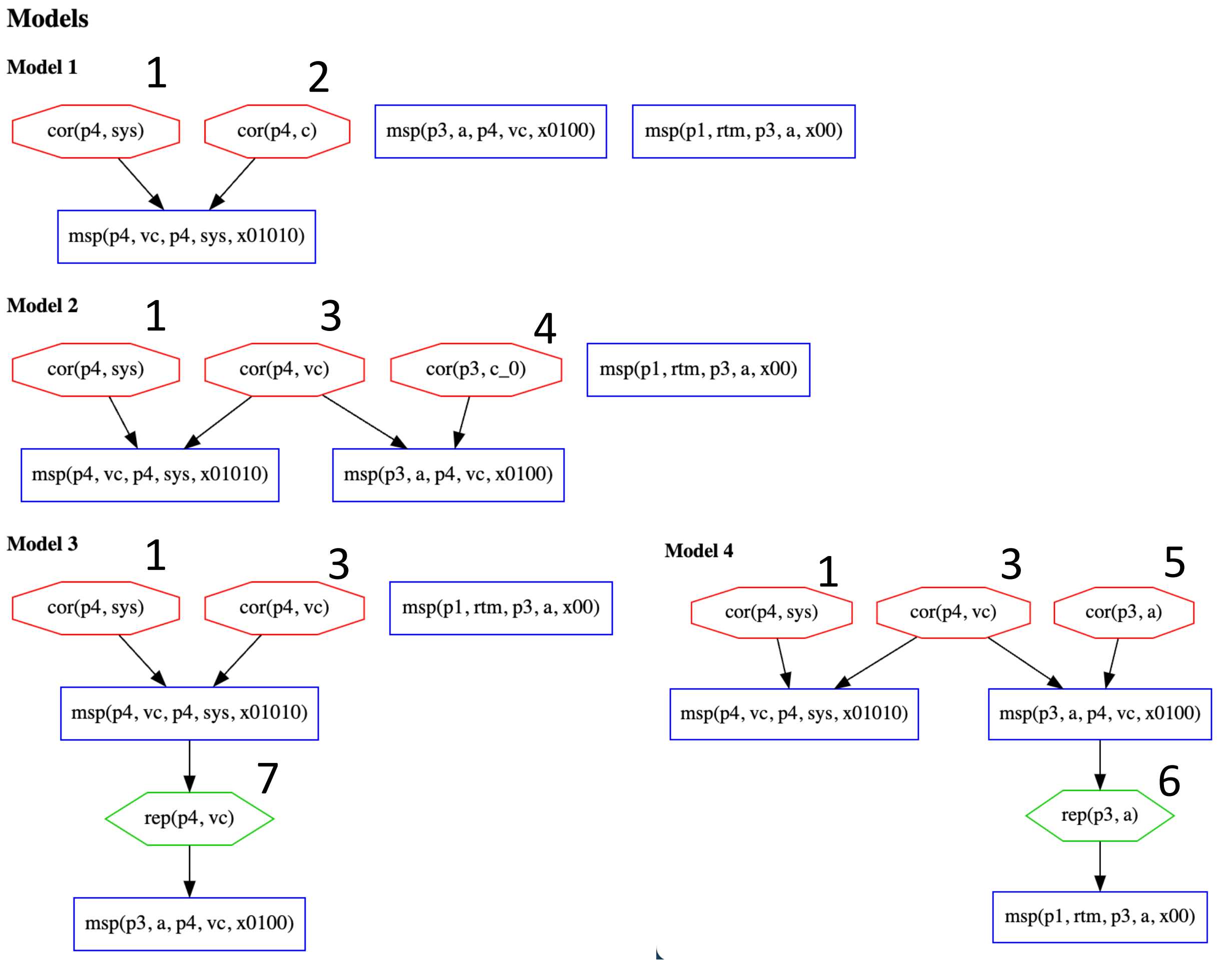


If you add deep thm about p3 model 2 and 3 are removed a-vc-sys-par

Measure *a* then *vc* then *sys* in parallel

- Protocol
  - *target: @p1 [rtm p3 *a*
    +~+ @p3 [a p4 *vc*
    +~+ @p4 [vc p4 *sys*]]]]

Model	Total cost
1	c1 + c2
2	c1 + c2 + c4
3	c1 + c3 + c7
4	c1 + c3 + c5 + c6

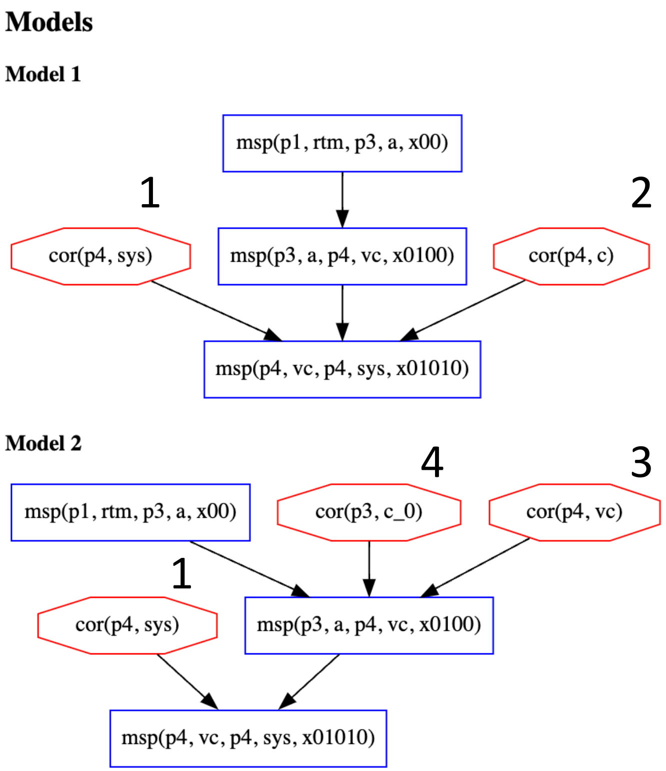


a-vc-sys-seq

Measure *a* then *vc* then *sys* in sequence

- Protocol
  - *target: @p1 [rtm p3 *a*
    +<+ @p3 [a p4 *vc*
    +<+ @p4 [vc p4 *sys*]]]]

Model	Total cost
1	c1 + c2
2	c1 + c3 + c4



All together

label	protocol	total cost
sys	*target: @p4 [vc p4 sys1]	(c1 + c3) OR (c1 + c2)
vc-sys-par	*target: @p3 [a p4 vc] +~+ @p4 [vc p4 sys]	(c1 + c2) OR (c1 + c3 + c5) OR (c1 + c3 + c4) OR (c1 + c3 + c7)
vc-sys-seq	*target: @p3 [a p4 vc] +<+ @p4 [vc p4 sys]	(c1 + c2) OR (c1 + c3 + c4) OR (c1 + c5 + c3)
a-vc-sys-par	*target: @p1 [rtm p3 *a* +~+ @p3 [a p4 *vc* +~+ @p4 [vc p4 *sys*]]]]	(c1 + c2) OR (c1 + c2 + c4) OR (c1 + c3 + c7) OR (c1 + c3 + c5 + c6)
a-vc-sys-seq	*target: @p1 [rtm p3 *a* +<+ @p3 [a p4 *vc* +<+ @p4 [vc p4 *sys*]]]]	(c1 + c2) OR (c1 + c3 + c4)

Event with label and cost

Event	Label	Cost	Present In
cor(p4,sys)	1	c1	sys(1,2),vc-sys-par(1,2,3,4), vc-sys-seq(1,2,3), a-vc-sys-par(1,2,3,4), a-vc-sys-seq(1,2)
cor(p4,c)	2	c2	sys(2), vc-sys-par(2), vc-sys-seq(1), a-vc-sys-par(1,2), a-vc-sys-seq(1)
cor(p4,vc)	3	c3	sys(1), vc-sys-par(2,3,4), vc-sys-seq(2,3), a-vc-sys-par(3,4), a-vc-sys-seq(2)
cor(p3, c_0)	4	c4	vc-sys-par(3), vc-sys-seq(2), a-vc-sys-par(2), a-vc-sys-seq(2)
cor(p3,a)	5	c5	vc-sys-par(2), vc-sys-seq(3), a-vc-sys-par(4)
rep(p3,a)	6	c6	a-vc-sys-par(4)
rep(p4,vc)	7	c7	vc-sys-par(4), a-vc-sys-par(3)

Thoughts/Takeaways