Владимир Рачкин и Павел Балай

Построение TLA$^+$ модели для смарт-контракта Phoenix Vault

Проект по курсу математической логики, весна 2022

04.06.2022

Факультет математики и компьютерных наук СПбГУ
Программа «Современное программирование»

Постановка задачи

1. Изучить смарт-контракт, описанный в статье[1]
2. Построить его TLA+ модель и проверить свойства
3. Смоделировать специальные события: потери ключа и атаки злоумышленника

[1] Phoenix: A Formally Verified Regenerating Vault (2021) за авторством Uri Kirstein, Shelly Grossman, Michael Mirkin, James Wilcox, Ittay Eyal, Mooly Sagiv

Что такое Phoenix Vault

Это обёртка для Etherium-кошелька

- 2 тира ключей: обычные - для отправки средств, привилегированные - для управления контрактом
- Возможность создавать новые ключи и удалять старые
- Задержка перед отправкой средств
- Возможность отменить отправку средств
- Возможность заблокировать отправку средств

Модель

Action	Key
Deposit	
Request	T_2
Withdraw	
Cancel request	T_1
Cancel all requests	T_1
Cancel self request	T_2
Lock	T_1
Add a T_1 key	T_1
Add a T_2 key	T_1
Remove a T_2 key	T_1

Events:
Tier-1 Key Loss
Tier-2 Key Loss
Type-1 Attack
Type-2 Attack

Состояния модели

- balance
- block_number
- tier_one_addresses
- tier_two_addresses
- delay
- unlock_block
- requests

Хранение предыдущего действия

```
Lock(address1, new_unlock_block) ==
    /\ previous_command' = (<<"lock", address1>>)
  ...
    /\ unlock_block' = new_unlock_block
    /\ UNCHANGED <<balance, tier_one_addresses, tier_two_addresses, delay,
       requests, special_vars>>

OnlyTierOneCanLock ==
    [][previous_command'[1] = "lock" => previous_command'[2] \in
       tier_one_addresses]_previous_command
```

МКН

Достижимость состояния

- A

Достижимость состояния

- A
- $\neg A$

МКН

Достижимость состояния

- A
- $\neg A$
- $\Box(\neg A)$

Достижимость состояния

- A
- $\neg A$
- $\Box(\neg A)$
- $\neg\Box(\neg A)$

Достижимость состояния

- A
- $\neg A$
- $\Box(\neg A)$
- $\neg\Box(\neg A)$
- $B \Rightarrow \neg\Box(\neg A)$

МКН

Достижимость состояния

- A
- $\neg A$
- $\Box(\neg A)$
- $\neg\Box(\neg A)$
- $B \Rightarrow \neg\Box(\neg A)$
- $\Box(B \Rightarrow \neg\Box(\neg A))$

Достижимость состояния 2

```
TierOneCanCancelAnyRequestAnyTime ==
[]((requests /= {} /\ block_number < MAX_BLOCK_NUMBER) =>
        LET b == block_number IN
            (\A r \in request_type:
                r \in requests =>
                (~[](~(
                    /\ block_number = b + 1
                    /\ previous_command[1] = "cancel_request"
                    /\ previous_command[2] \in tier_one_addresses
                    /\ previous_command[3] = r[1])))))
```

Использование ‘ENABLED‘

```
TierOneCanCancelAnyRequestAnyTime ==
    [](block_number < MAX_BLOCK_NUMBER
        => \A <<address1, req>> \in tier_one_addresses \X requests:
        ENABLED CancelRequest(address1, req[1]))
```

МКН

Изменение модели

Добавили 2 множества ключей о которых знает владелец и злоумышленник

Добавили в Actions запуск событий и разделили действия владельца и злоумышленника

Добавили свойства, гарантирующие обработку всех событий

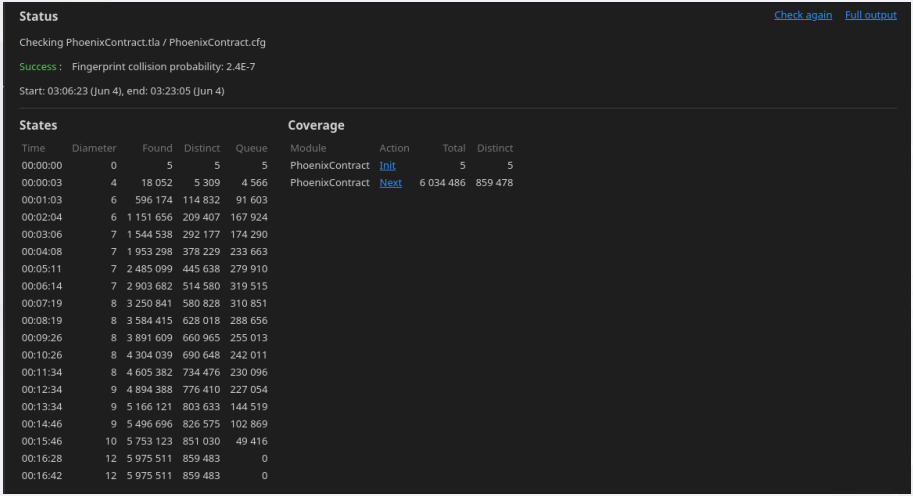
Безопасность

```
Defence ==
    \/ TierOneLossDefence
    \/ TierTwoLossDefence
    \/ TypeOneAttackDefence
    \/ TypeTwoAttackDefence

Next ==
    IF ENABLED Defence
    THEN Defence
    ELSE Actions \/ ActionTick
```

МКН

Проверка модели



Status Check again Full output

Checking PhoenixContract.tla / PhoenixContract.cfg

Success : Fingerprint collision probability: 2.4E-7

Start: 03:06:23 (Jun 4), end: 03:23:05 (Jun 4)

States

Time	Diameter	Found	Distinct	Queue
00:00:00	0	5	5	5
00:00:03	4	18 052	5 309	4 566
00:01:03	6	596 174	114 832	91 603
00:02:04	6	1 151 656	209 407	167 924
00:03:06	7	1 544 538	292 177	174 290
00:04:08	7	1 953 298	378 229	233 663
00:05:11	7	2 485 099	445 638	279 910
00:06:14	7	2 903 682	514 580	319 515
00:07:19	8	3 250 841	580 828	310 851
00:08:19	8	3 584 415	628 018	288 656
00:09:26	8	3 891 609	660 965	255 013
00:10:26	8	4 304 039	690 648	242 011
00:11:34	8	4 605 382	734 476	230 096
00:12:34	9	4 894 388	776 410	227 054
00:13:34	9	5 166 121	803 633	144 519
00:14:46	9	5 496 696	826 575	102 869
00:15:46	10	5 753 123	851 030	49 416
00:16:28	12	5 975 511	859 483	0
00:16:42	12	5 975 511	859 483	0

Coverage

Module	Action	Total	Distinct
PhoenixContract	Init	5	5
PhoenixContract	Next	6 034 486	859 478

Результаты работы

1. Построена и проверена модель контракта
2. Смоделированы атаки и потери ключей

Владимир Рачкин @robozmey
Павел Балай @Koropok
Ссылка на Github проекта phoenix_proof