UNIVERSITÀ
DI PARMA

Dipartimento di Scienze Matematiche, Fisiche e Informatiche

Corso di Laurea Triennale in Informatica

Candidato
Saverio Mattia Merenda, 330503

Relatore
Prof. Vincenzo Arceri

Costruzione di Control-Flow Graph completi per bytecode EVM

Introduzione

UNIVERSITÀ DI PARMA

Blockchain

Introduzione

UNIVERSITÀ DI PARMA

Blockchain

Verifica della correttezza del codice

Introduzione

UNIVERSITÀ DI PARMA

Blockchain

Verifica della correttezza del codice

Control-Flow Graph

Introduzione

UNIVERSITÀ DI PARMA

Blockchain

Verifica della correttezza del codice

Control-Flow Graph

Interpretazione astratta

Introduzione

UNIVERSITÀ DI PARMA

Blockchain

Verifica della correttezza del codice

Control-Flow Graph

Interpretazione astratta

Introduzione alla blockchain

UNIVERSITÀ DI PARMA

Introduzione alla blockchain

UNIVERSITÀ DI PARMA

Satoshi Nakamoto, 2008

Introduzione alla blockchain

Satoshi Nakamoto, 2008

Registro digitale

Introduzione alla blockchain

Satoshi Nakamoto, 2008

Registro digitale

Distribuito

Introduzione alla blockchain

Satoshi Nakamoto, 2008

Registro digitale

Distribuito

Catena di blocchi

Introduzione alla blockchain

Satoshi Nakamoto, 2008

Registro digitale

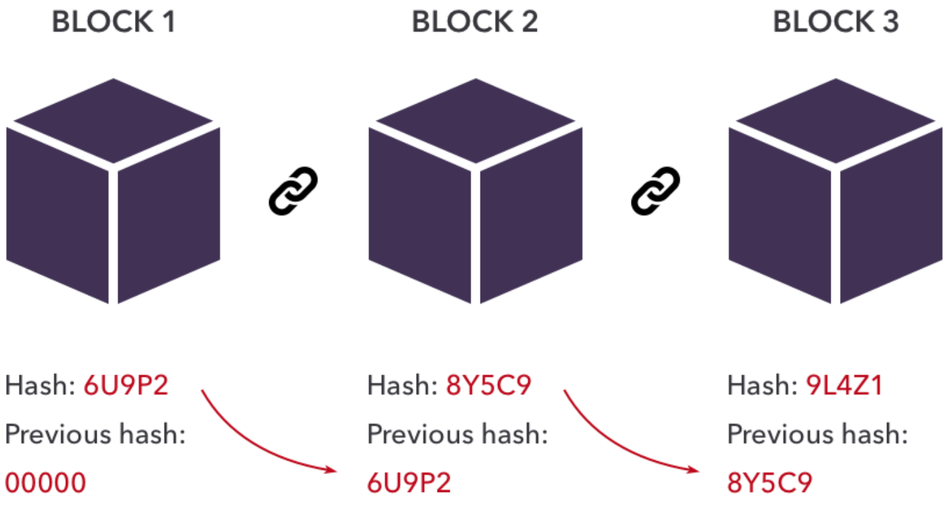
Distribuito

Catena di blocchi

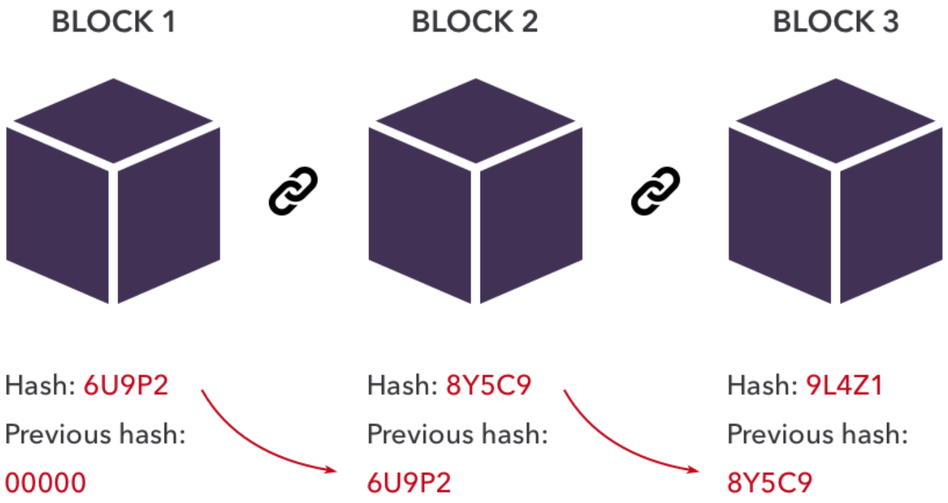
Sicuro e affidabile

Funzionamento blockchain

UNIVERSITÀ DI PARMA

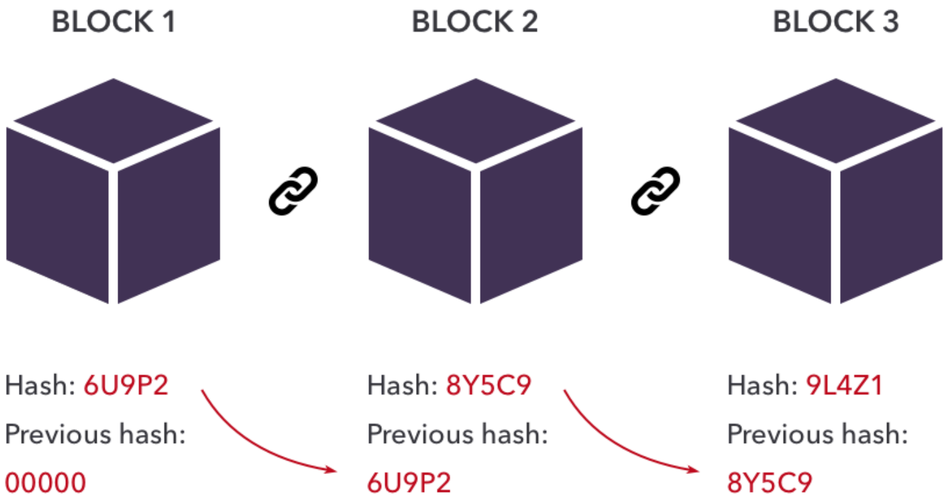


Funzionamento blockchain



Blocchi interconnessi

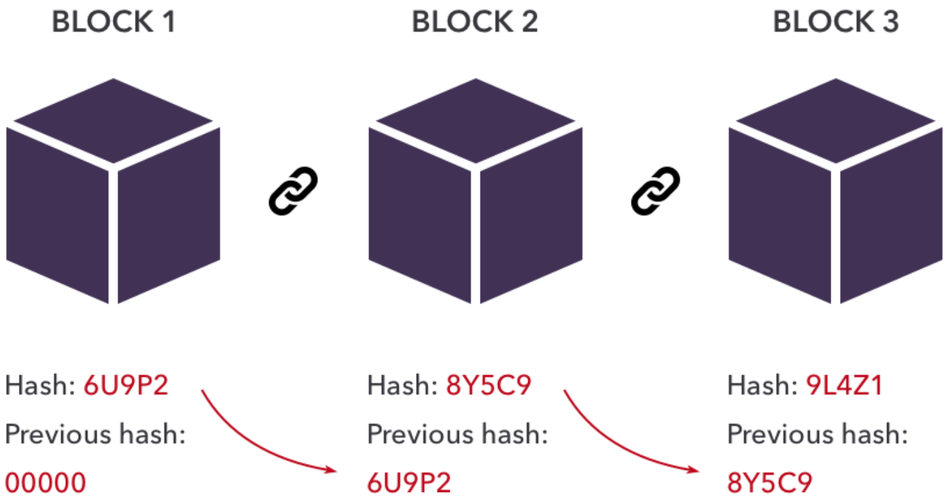
Funzionamento blockchain



Blocchi interconnessi

Crittografia

Funzionamento blockchain

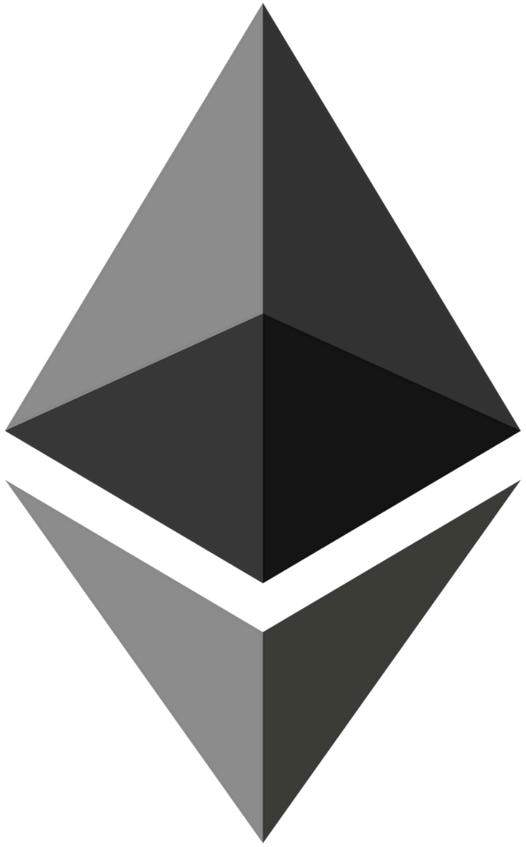


Blocchi interconnessi

Crittografia

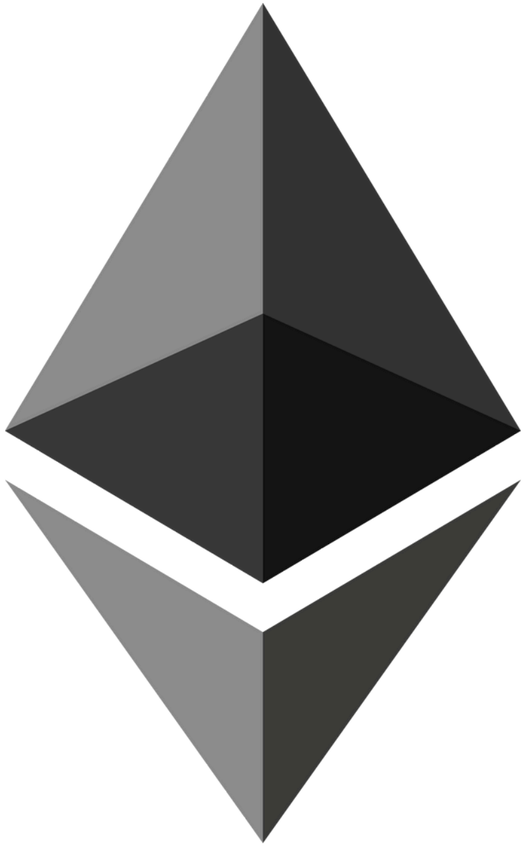
Catena inalterabile

Ethereum



Vitalik Buterin, 2013

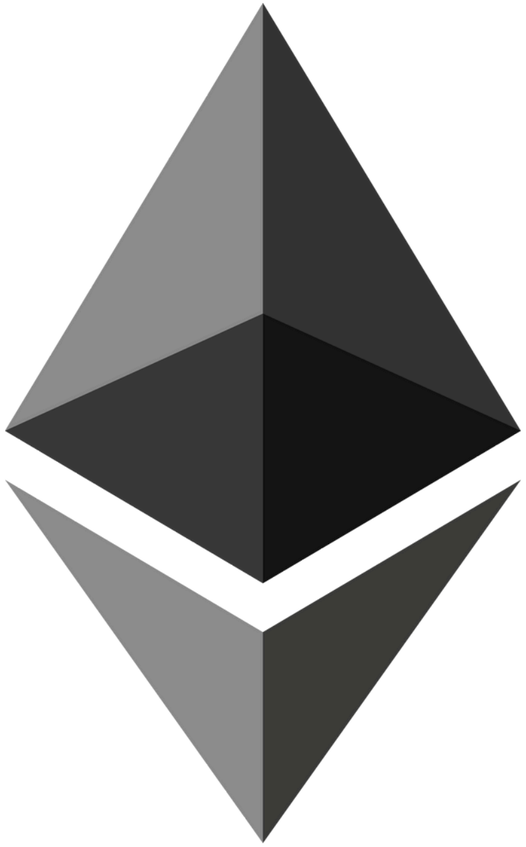
Ethereum



Vitalik Buterin, 2013

Pubblica & open-source

Ethereum

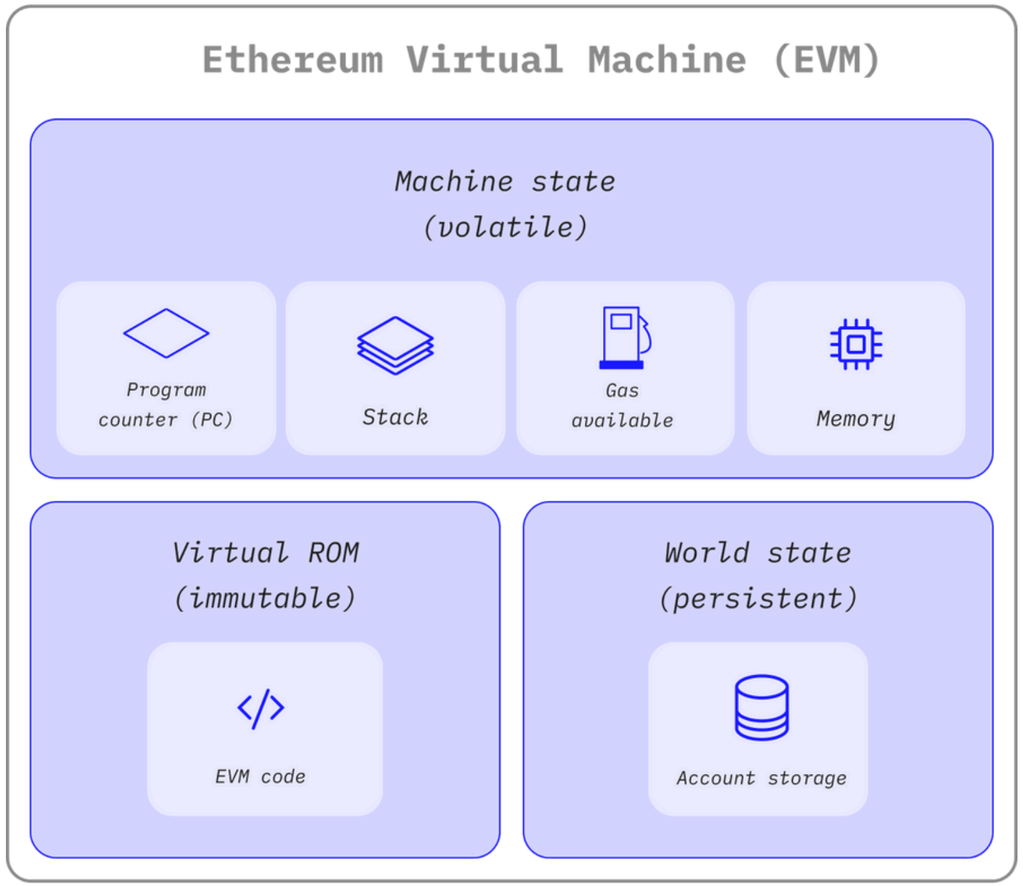


Vitalik Buterin, 2013

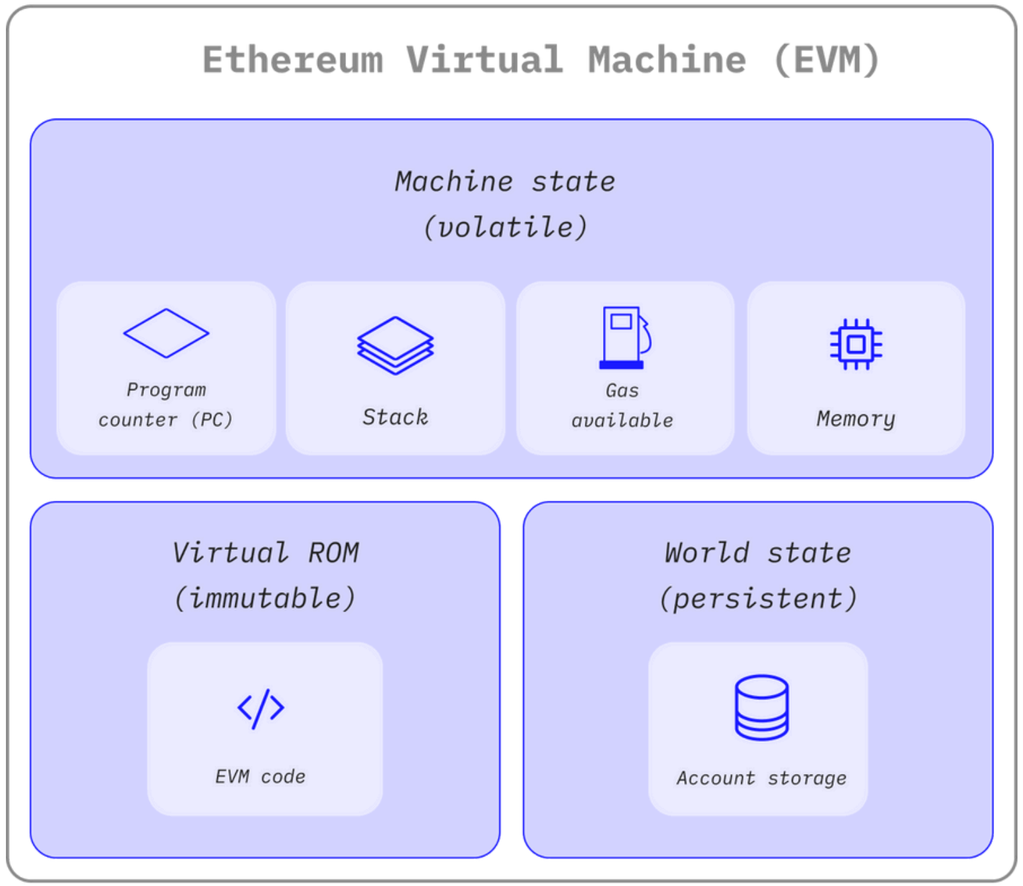
Pubblica & open-source

Smart contract

Ethereum Virtual Machine

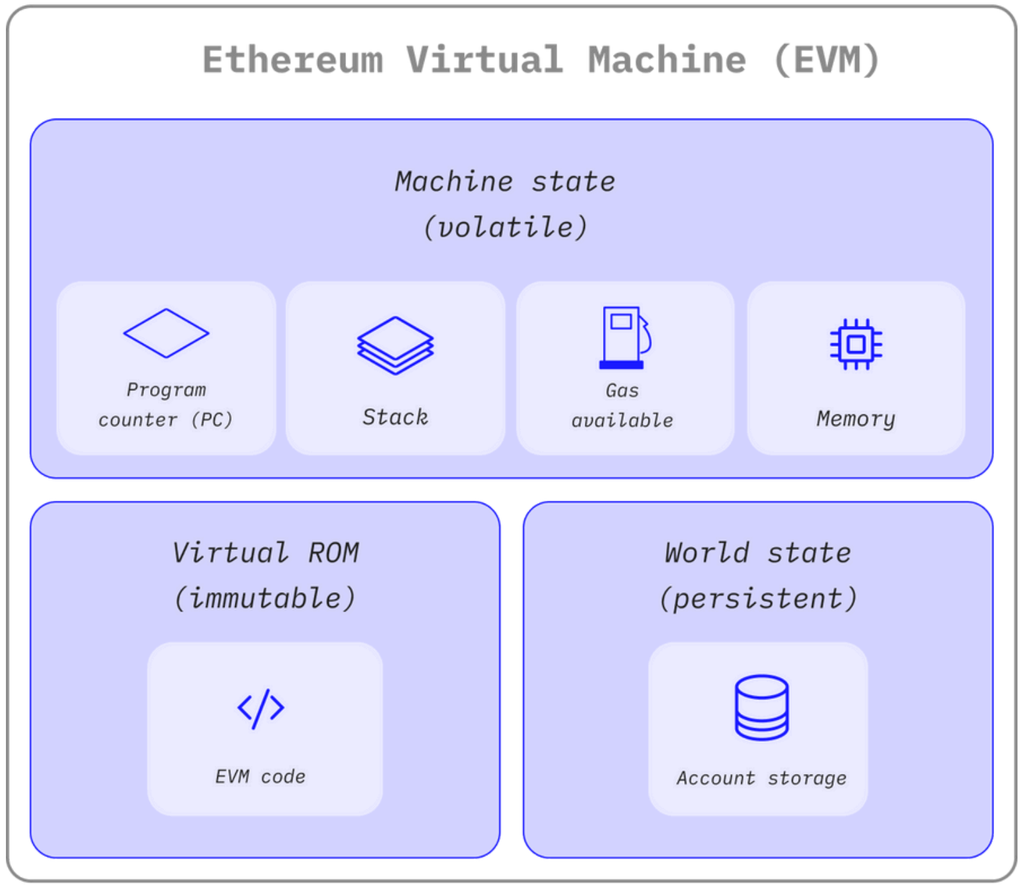


Ethereum Virtual Machine



Cuore di Ethereum

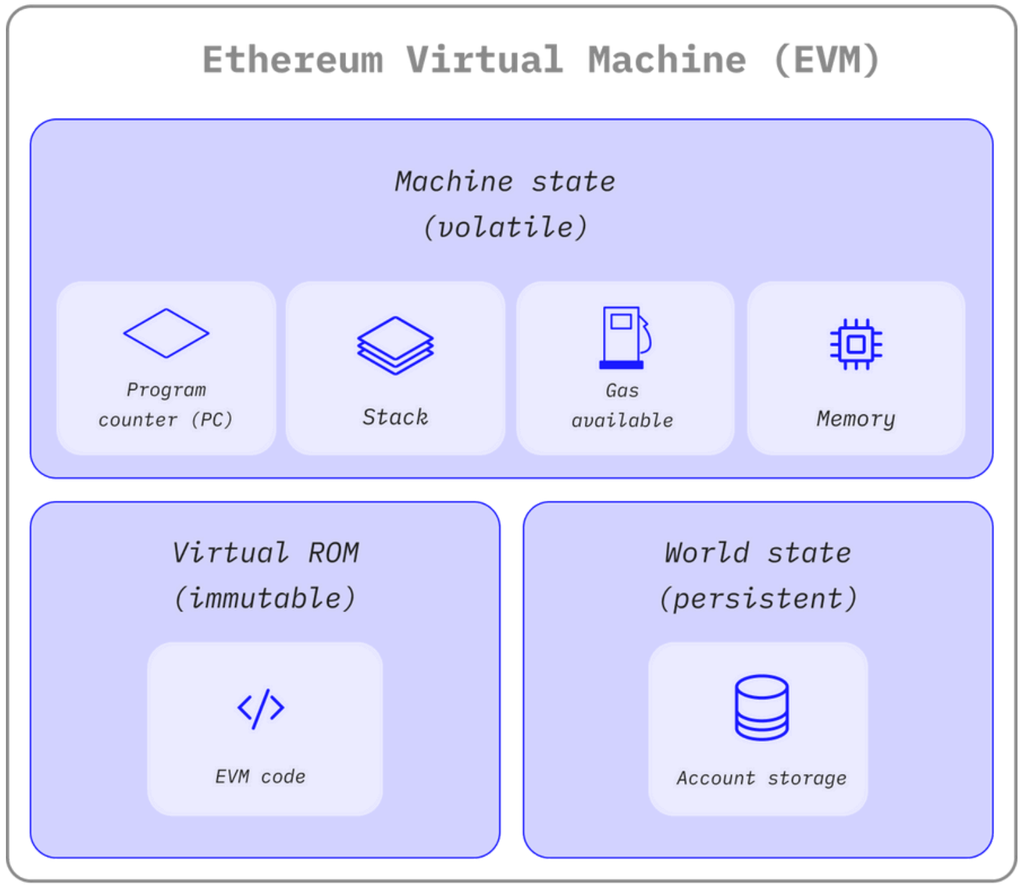
Ethereum Virtual Machine



Cuore di Ethereum

Gestione smart contract

Ethereum Virtual Machine

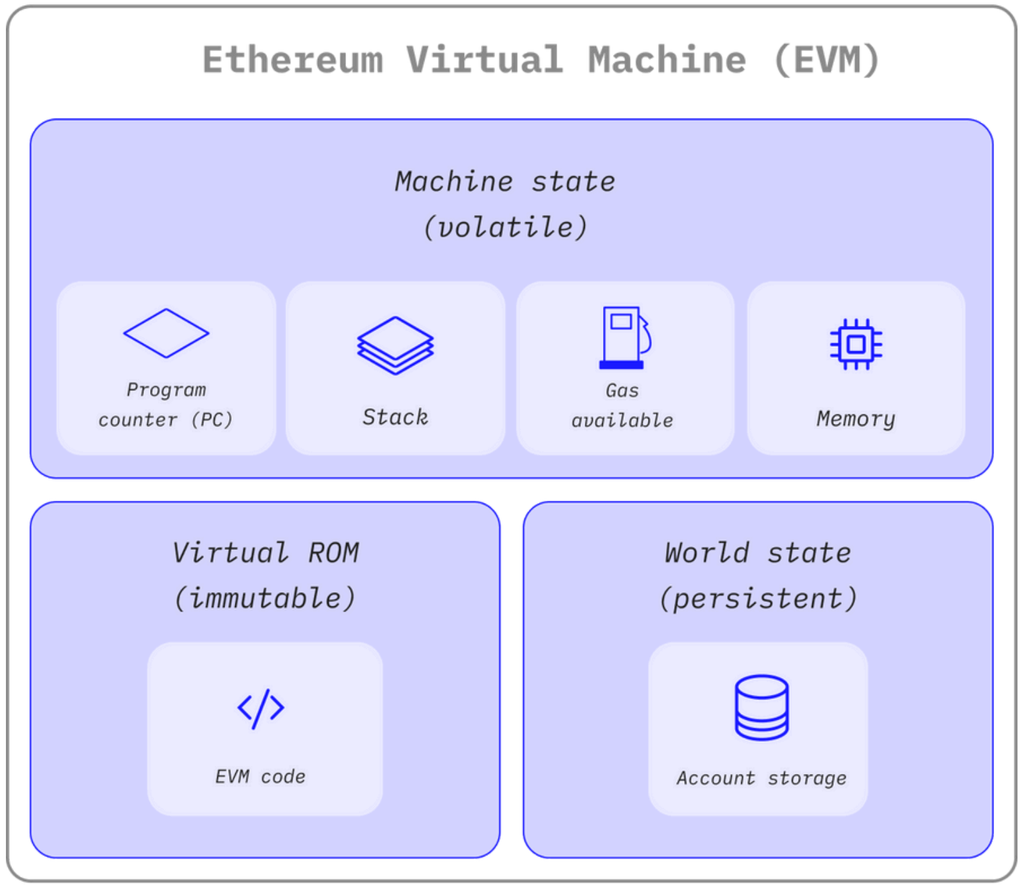


Cuore di Ethereum

Gestione smart contract

Funzione matematica

Ethereum Virtual Machine



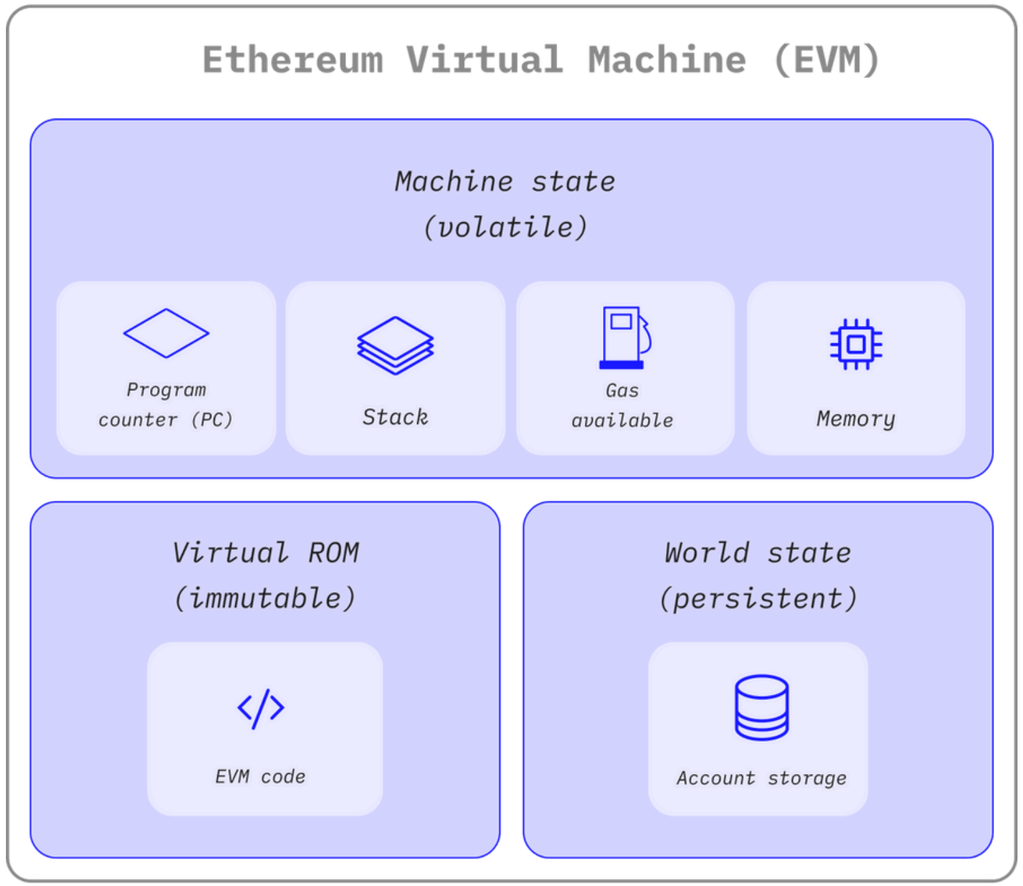
Cuore di Ethereum

Gestione smart contract

Funzione matematica

Stack, Memory & Storage

Ethereum Virtual Machine



Cuore di Ethereum

Gestione smart contract

Funzione matematica

Stack, Memory & Storage

Terminazione garantita

EVM Bytecode

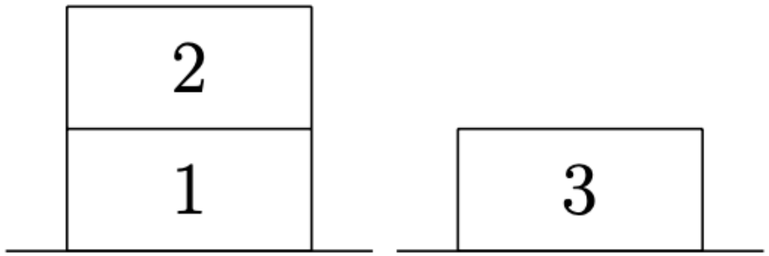
UNIVERSITÀ DI PARMA

```
PUSH1 0x01
PUSH1 0x02
ADD
```

Basso livello

EVM Bytecode

```
PUSH1 0x01
PUSH1 0x02
ADD
```

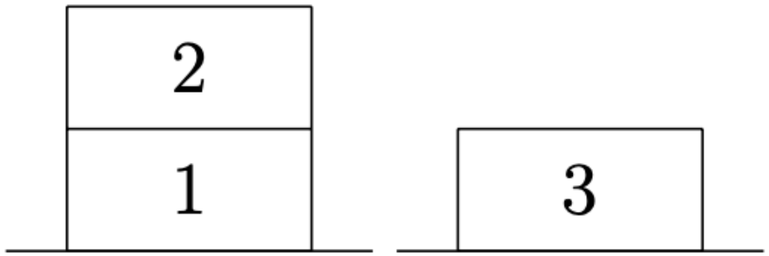


Basso livello

Basato su stack

EVM Bytecode

```
PUSH1 0x01
PUSH1 0x02
ADD
```



Basso livello

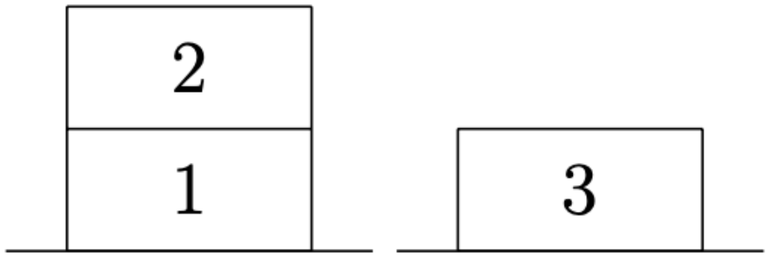
Basato su stack

Oltre 150 opcode

EVM Bytecode

UNIVERSITÀ DI PARMA

```
PUSH1 0x01
PUSH1 0x02
ADD
```



Basso livello

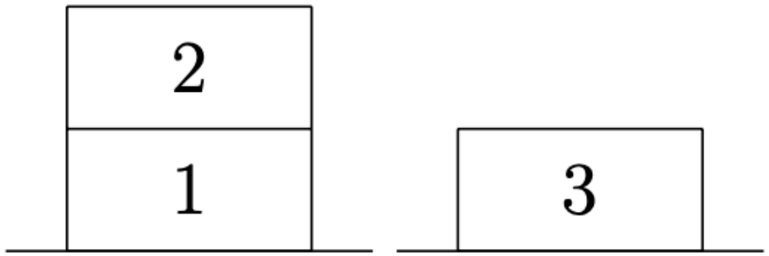
Basato su stack

Oltre 150 opcode

Molte operazioni

EVM Bytecode

```
PUSH1 0x01
PUSH1 0x02
ADD
```



Basso livello

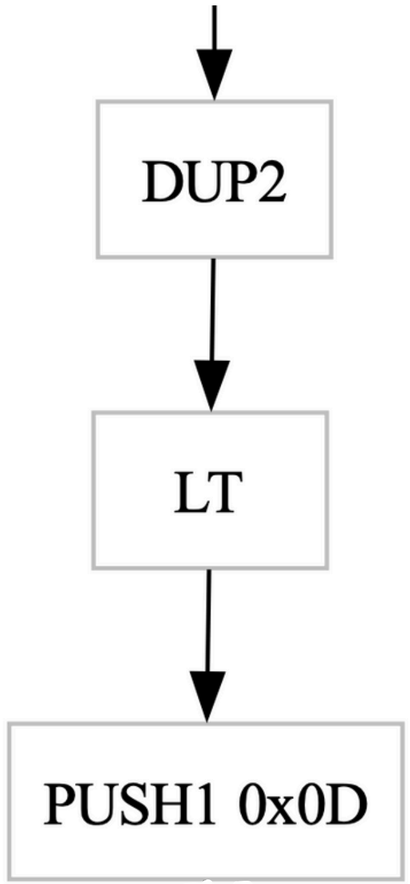
Basato su stack

Oltre 150 opcode

Molte operazioni

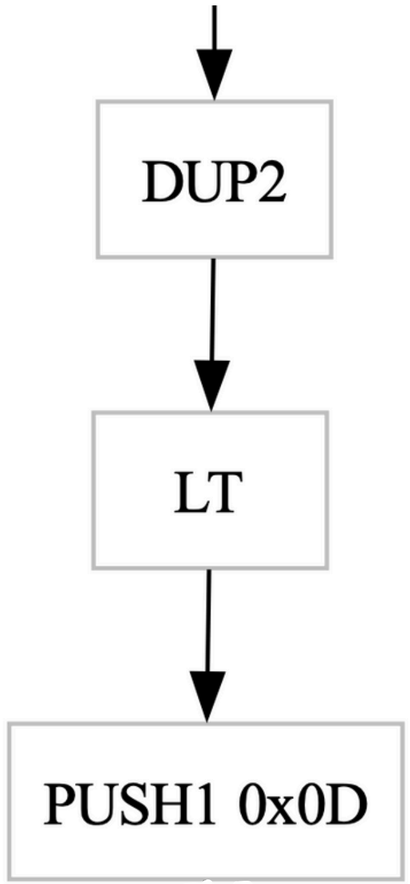
Accesso ad informazioni

Flusso di esecuzione



Flusso sequenziale

Flusso di esecuzione

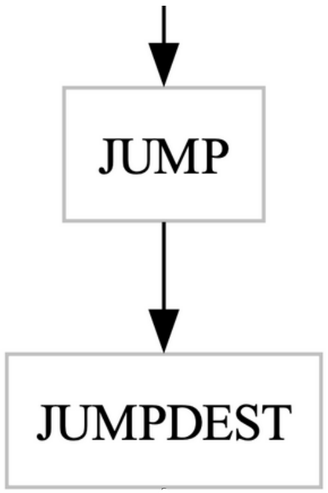


Flusso sequenziale

JUMP & JUMPI

Flusso di esecuzione

```
PUSH1 0x01
PUSH1 0x02
JUMP
```



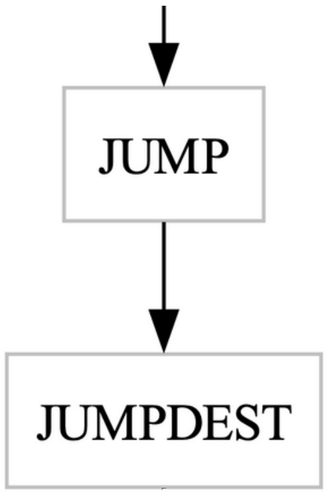
Flusso sequenziale

JUMP & JUMPI

Salto incondizionato

Flusso di esecuzione

```
PUSH1 0x01
PUSH1 0x02
JUMP
```



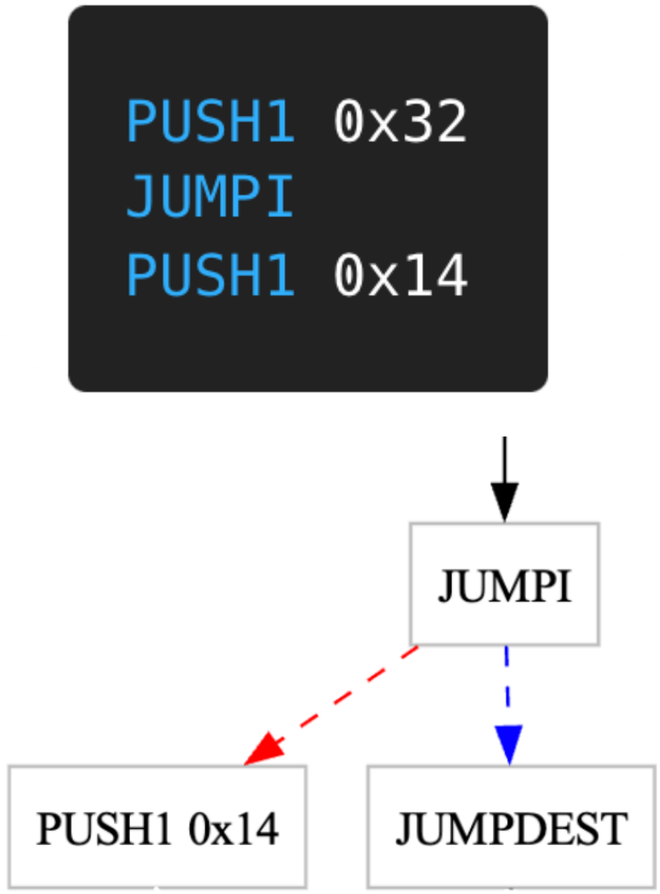
Flusso sequenziale

JUMP & JUMPI

Salto incondizionato

JUMPDEST

Flusso di esecuzione



Flusso sequenziale

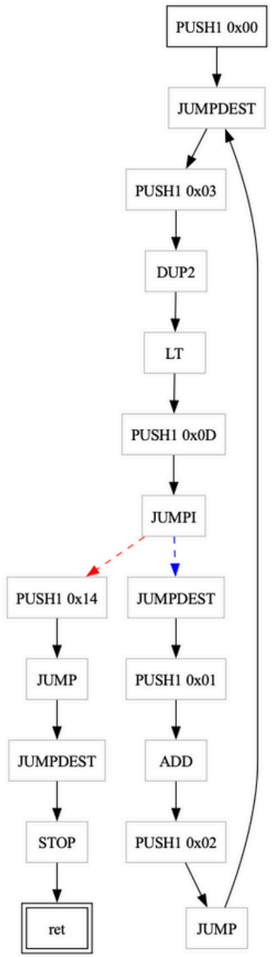
JUMP & JUMPI

Salto incondizionato

JUMPDEST

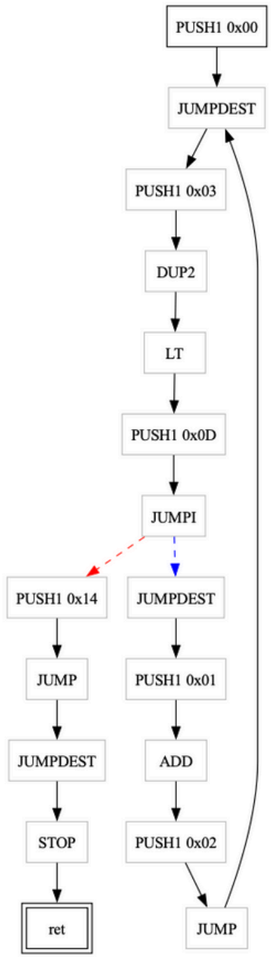
Salto condizionato

Costruire un CFG completo



Control-flow Graph

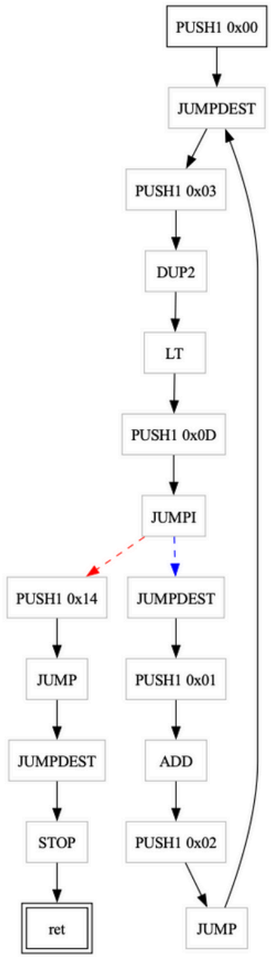
Costruire un CFG completo



Control-flow Graph

Grafo dei possibili percorsi di esecuzione

Costruire un CFG completo

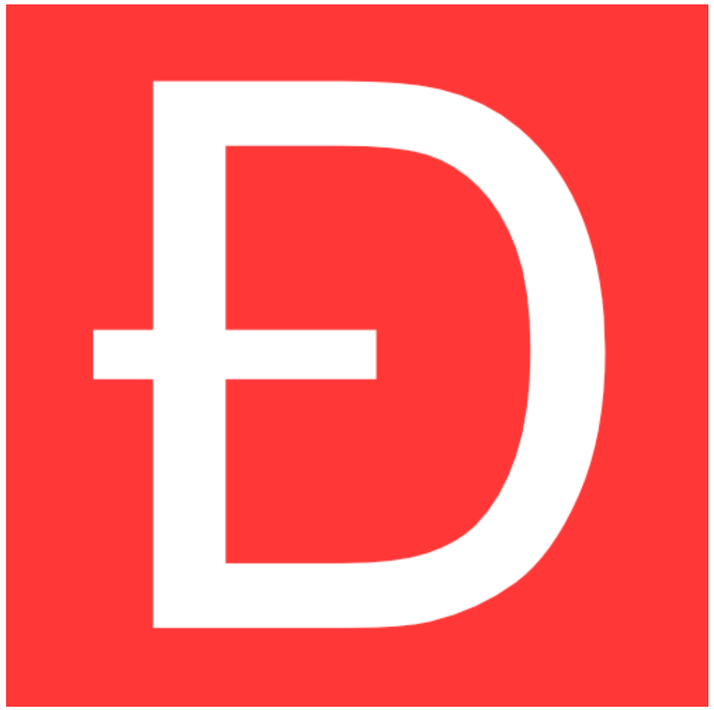


Control-flow Graph

Grafo dei possibili percorsi di esecuzione

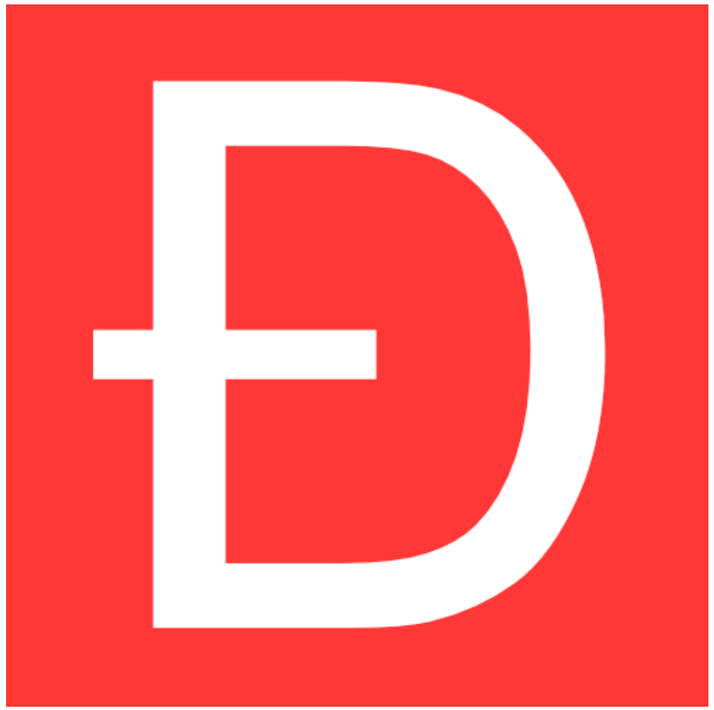
Sound per un’analisi corretta

The DAO attack



2016

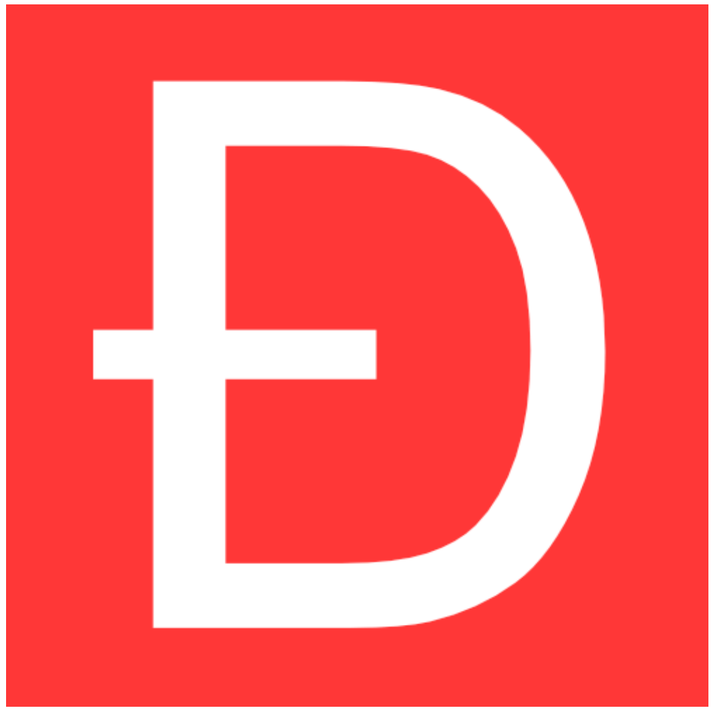
The DAO attack



2016

Episodio significativo

The DAO attack

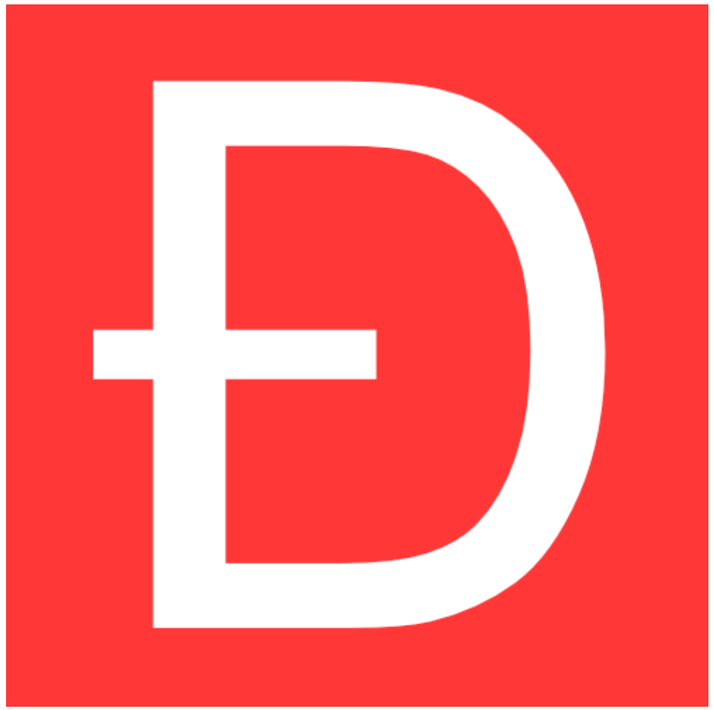


2016

Episodio significativo

3.6 milioni di *ether*

The DAO attack



2016

Episodio significativo

3.6 milioni di *ether*

$60 milioni all’epoca

Problemi principali

UNIVERSITÀ DI PARMA

JUMP & JUMPI

Problemi principali

UNIVERSITÀ DI PARMA

```
PUSH1 0x01
PUSH1 0x02
JUMP
```

JUMP & JUMPI

Pushed jumps

Problemi principali

UNIVERSITÀ DI PARMA

```
PUSH1 0x0A
PUSH1 0x0C
ADD
JUMP
```

JUMP & JUMPI

Pushed jumps

Orphan jumps

Problemi principali

UNIVERSITÀ DI PARMA

```
PUSH1 0x0A
PUSH1 0x0C
ADD
JUMP
```

JUMP & JUMPI

Pushed jumps

Orphan jumps

Vulnerabilità ad attacchi

UNIVERSITÀ DI PARMA

Smart contract sono software

Analisi del software

UNIVERSITÀ DI PARMA

Smart contract sono software

Soggetti a vulnerabilità ed errori

Analisi del software

UNIVERSITÀ DI PARMA

Smart contract sono software

Soggetti a vulnerabilità ed errori

Analisi statica

Analisi del software

UNIVERSITÀ DI PARMA

Smart contract sono software

Soggetti a vulnerabilità ed errori

Analisi statica

Correttezza senza eseguirlo

UNIVERSITÀ DI PARMA

Teoria formale

Interpretazione astratta

UNIVERSITÀ DI PARMA

Teoria formale

Approssima oggetti matematici

Interpretazione astratta

Teoria formale

Approssima oggetti matematici

Relazioni

Interpretazione astratta

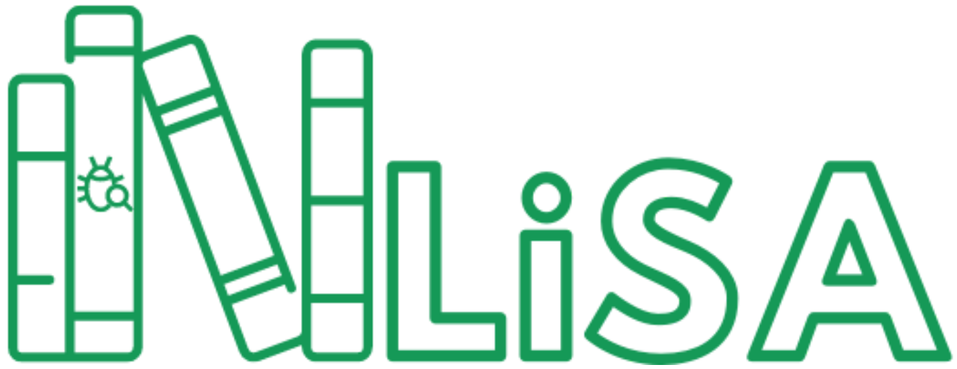
UNIVERSITÀ DI PARMA

Teoria formale

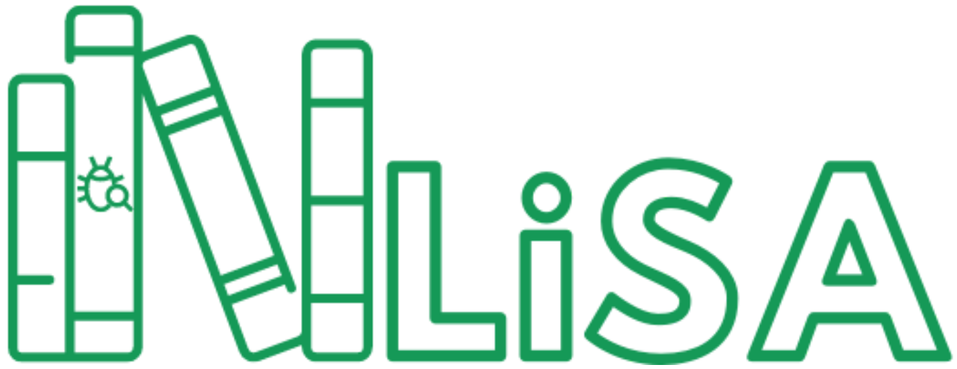
Approssima oggetti matematici

Relazioni

Approssimare comportamenti concreti lavorando sull’astrazione

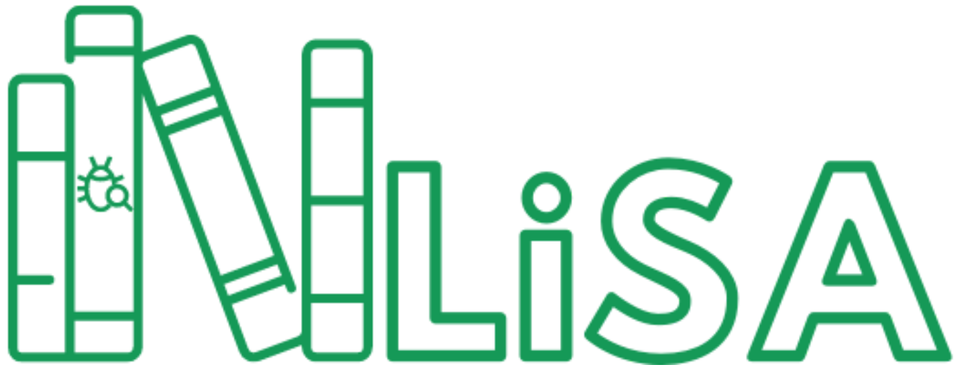


Library for Static Analysis



Library for Static Analysis

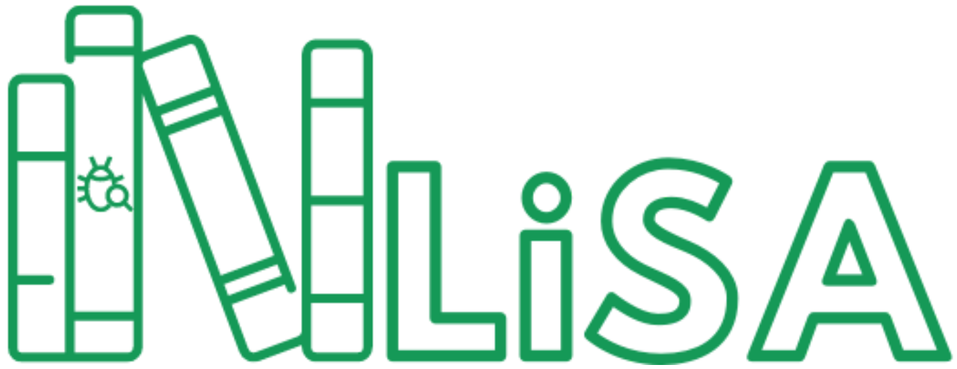
Analizzatori statici



Library for Static Analysis

Analizzatori statici

CFG di un programma



Library for Static Analysis

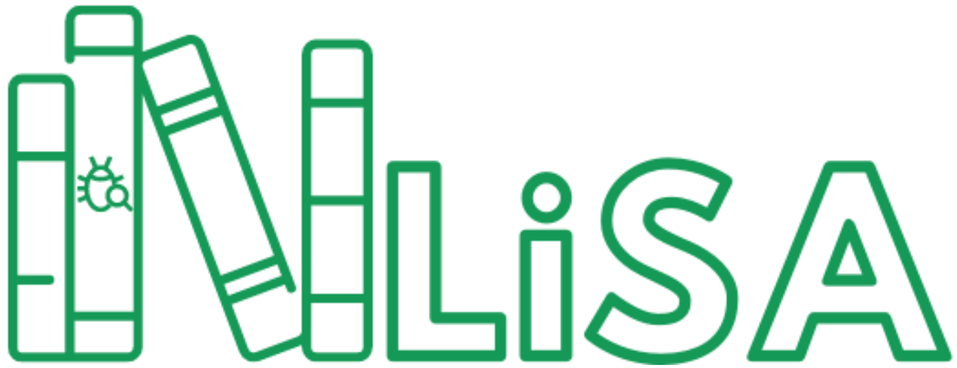
Analizzatori statici

CFG di un programma

Verifica delle proprietà

LiSA & EVMLiSA

UNIVERSITÀ DI PARMA



Library for Static Analysis

Analizzatori statici

CFG di un programma

Verifica delle proprietà

EVM Bytecode

Legge il bytecode EVM

CFG in EVMLiSA

UNIVERSITÀ DI PARMA

Legge il bytecode EVM

Analizza il flusso di esecuzione

CFG in EVMLiSA

UNIVERSITÀ DI PARMA

Legge il bytecode EVM

Analizza il flusso di esecuzione

JUMP genera un *Sequential edge*

CFG in EVMLiSA

UNIVERSITÀ DI PARMA

Legge il bytecode EVM

Analizza il flusso di esecuzione

JUMP genera un *Sequential edge*

JUMPI genera un *True edge* e un *False edge*

CFG in EVMLiSA

UNIVERSITÀ DI PARMA

Legge il bytecode EVM

Analizza il flusso di esecuzione

JUMP genera un *Sequential edge*

JUMPI genera un *True edge* e un *False edge*

JUMPDEST

Domini astratti

UNIVERSITÀ DI PARMA

EVM usa stack volatile per elaborare istruzioni

Domini astratti

EVM usa stack volatile per elaborare istruzioni

Stack astratto

Domini astratti

EVM usa stack volatile per elaborare istruzioni

Stack astratto

Elementi dello stack astratto

Elementi dello stack astratto

$$\mathsf{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\mathsf{Ints}_k}\}, \sqcup_{\mathsf{Ints}_k}, \sqcap_{\mathsf{Ints}_k}, \top_{\mathsf{Ints}_k}, \varnothing \rangle$$

Elementi dello stack astratto

$$\mathsf{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\mathsf{Ints}_k}\}, \sqcup_{\mathsf{Ints}_k}, \sqcap_{\mathsf{Ints}_k}, \top_{\mathsf{Ints}_k}, \varnothing \rangle$$

Insieme di k numeri interi

Elementi dello stack astratto

$$\mathsf{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\mathsf{Ints}_k}\}, \sqcup_{\mathsf{Ints}_k}, \sqcap_{\mathsf{Ints}_k}, \top_{\mathsf{Ints}_k}, \varnothing \rangle$$

Insieme di k numeri interi

Parzialmente ordinati

Elementi dello stack astratto

UNIVERSITÀ DI PARMA

$$\mathsf{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\mathsf{Ints}_k}\}, \sqcup_{\mathsf{Ints}_k}, \sqcap_{\mathsf{Ints}_k}, \top_{\mathsf{Ints}_k}, \varnothing \rangle$$

Insieme di k numeri interi

Parzialmente ordinati

$\varnothing$ elemento inferiore (errore / irraggiungibile)

Elementi dello stack astratto

UNIVERSITÀ DI PARMA

$$\mathsf{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\mathsf{Ints}_k}\}, \sqcup_{\mathsf{Ints}_k}, \sqcap_{\mathsf{Ints}_k}, \top_{\mathsf{Ints}_k}, \varnothing \rangle$$

Insieme di k numeri interi

Parzialmente ordinati

$\varnothing$ elemento inferiore (errore / irraggiungibile)

$\top_{\mathsf{Ints}_k}$ valore non rappresentabile

Stack astratto

$$\mathcal{S}_{\mathsf{Ints}_k,h} \triangleq \{[s_0, s_1, \ldots, s_{h-1}] \mid \forall i \in [0, h-1] \,.\, s_i \in \mathsf{Ints}_k, h, k > 0\}$$

Stack astratto

$$\mathcal{S}_{\mathsf{Ints}_k,h} \triangleq \{[s_0, s_1, \ldots, s_{h-1}] \mid \forall i \in [0, h-1] \,.\, s_i \in \mathsf{Ints}_k, h, k > 0\}$$

Contiene i primi h elementi Ints_k dello stack

Stack astratto

$$\mathcal{S}_{\mathsf{Ints}_k,h} \triangleq \{[s_0, s_1, \ldots, s_{h-1}] \mid \forall i \in [0, h-1] \,.\, s_i \in \mathsf{Ints}_k, h, k > 0\}$$

Contiene i primi h elementi Ints_k dello stack

$$\mathsf{St}^{\#}_{k,h} \triangleq \langle \mathcal{S}_{\mathsf{Ints}_k,h} \cup \{\bot_{\mathsf{St}^{\#}_{k,h}}\}, \sqcup_{\mathsf{St}^{\#}_{k,h}}, \sqcap_{\mathsf{St}^{\#}_{k,h}}, \top_{\mathsf{St}^{\#}_{k,h}}, \bot_{\mathsf{St}^{\#}_{k,h}} \rangle$$

Stack astratto

$$\mathcal{S}_{\mathsf{Ints}_k,h} \triangleq \{[s_0, s_1, \ldots, s_{h-1}] \mid \forall i \in [0, h-1] \,.\, s_i \in \mathsf{Ints}_k, h, k > 0\}$$

Contiene i primi h elementi Ints_k dello stack

$$\mathsf{St}^{\#}_{k,h} \triangleq \langle \mathcal{S}_{\mathsf{Ints}_k,h} \cup \{\bot_{\mathsf{St}^{\#}_{k,h}}\}, \sqcup_{\mathsf{St}^{\#}_{k,h}}, \sqcap_{\mathsf{St}^{\#}_{k,h}}, \top_{\mathsf{St}^{\#}_{k,h}}, \bot_{\mathsf{St}^{\#}_{k,h}} \rangle$$

$\bot_{\mathsf{St}^{\#}_{k,h}}$ **elemento inferiore (*bottom*)**

Stack astratto

UNIVERSITÀ DI PARMA

$$\mathcal{S}_{\mathsf{Ints}_k,h} \triangleq \{[s_0, s_1, \ldots, s_{h-1}] \mid \forall i \in [0, h-1] \,.\, s_i \in \mathsf{Ints}_k, h, k > 0\}$$

Contiene i primi h elementi Ints_k dello stack

$$\mathsf{St}^{\#}_{k,h} \triangleq \langle \mathcal{S}_{\mathsf{Ints}_k,h} \cup \{\bot_{\mathsf{St}^{\#}_{k,h}}\}, \sqcup_{\mathsf{St}^{\#}_{k,h}}, \sqcap_{\mathsf{St}^{\#}_{k,h}}, \top_{\mathsf{St}^{\#}_{k,h}}, \bot_{\mathsf{St}^{\#}_{k,h}} \rangle$$

$\bot_{\mathsf{St}^{\#}_{k,h}}$ elemento inferiore (*bottom*)

$$\top_{\mathsf{St}^{\#}_{k,h}} = [\top_{\mathsf{Ints}_k}, \top_{\mathsf{Ints}_k}, \ldots, \top_{\mathsf{Ints}_k}]$$

Insieme degli stack astratti

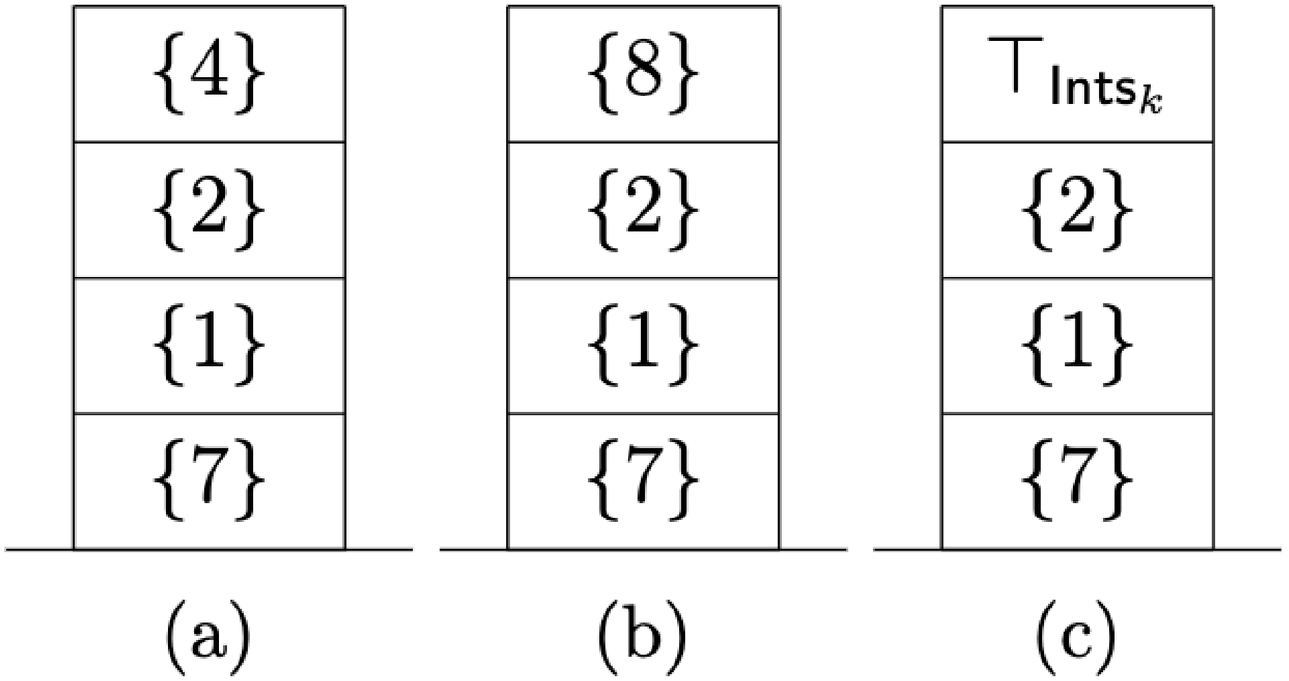
UNIVERSITÀ DI PARMA

Maggiore efficacia nella risoluzione delle jump

Insieme degli stack astratti

UNIVERSITÀ DI PARMA

Maggiore efficacia nella risoluzione delle jump



$$k = 1$$

Insieme degli stack astratti

UNIVERSITÀ DI PARMA

Maggiore efficacia nella risoluzione delle jump

$$\mathsf{SetSt}^{\#}_{k,h,l} \triangleq \langle \wp_{\leq l}(\mathcal{S}_{\mathsf{Ints}_k,h}) \cup \{\top_{\mathsf{SetSt}^{\#}_{k,h,l}}\}, \sqcup_{\mathsf{SetSt}^{\#}_{k,h,l}}, \sqcap_{\mathsf{SetSt}^{\#}_{k,h,l}}, \top_{\mathsf{SetSt}^{\#}_{k,h,l}}, \varnothing \rangle$$

Insieme degli stack astratti

UNIVERSITÀ DI PARMA

Maggiore efficacia nella risoluzione delle jump

$$\mathsf{SetSt}^{\#}_{k,h,l} \triangleq \langle \wp_{\leq l}(\mathcal{S}_{\mathsf{Ints}_k,h}) \cup \{\top_{\mathsf{SetSt}^{\#}_{k,h,l}}\}, \sqcup_{\mathsf{SetSt}^{\#}_{k,h,l}}, \sqcap_{\mathsf{SetSt}^{\#}_{k,h,l}}, \top_{\mathsf{SetSt}^{\#}_{k,h,l}}, \varnothing \rangle$$

Contiene al più l elementi $\mathcal{S}_{\mathsf{Ints}_k,h}$

Insieme degli stack astratti

UNIVERSITÀ DI PARMA

Maggiore efficacia nella risoluzione delle jump

$$\mathsf{SetSt}^{\#}_{k,h,l} \triangleq \langle \wp_{\leq l}(\mathcal{S}_{\mathsf{Ints}_k,h}) \cup \{\top_{\mathsf{SetSt}^{\#}_{k,h,l}}\}, \sqcup_{\mathsf{SetSt}^{\#}_{k,h,l}}, \sqcap_{\mathsf{SetSt}^{\#}_{k,h,l}}, \top_{\mathsf{SetSt}^{\#}_{k,h,l}}, \varnothing \rangle$$

Contiene al più l elementi $\mathcal{S}_{\mathsf{Ints}_k,h}$

***top* quando sforiamo la dimensione l**

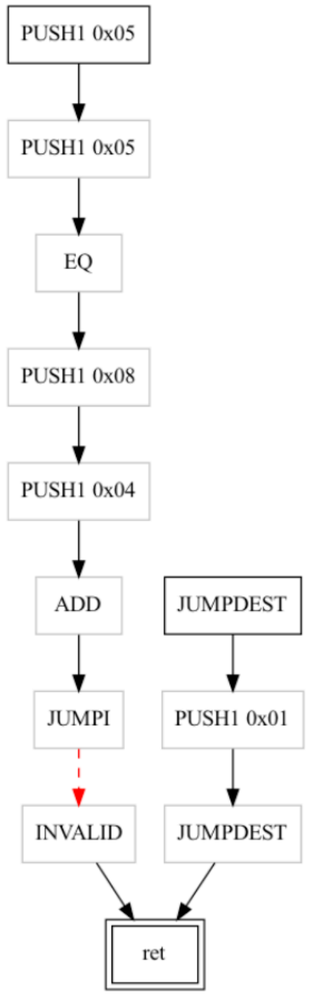
Risoluzione orphan jumps

UNIVERSITÀ DI PARMA

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```

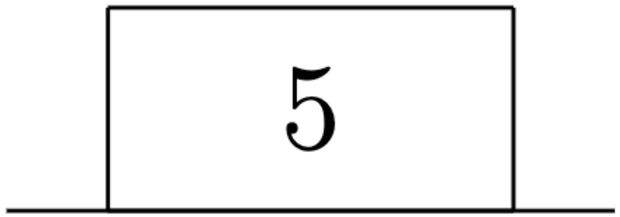
Risoluzione orphan jumps

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



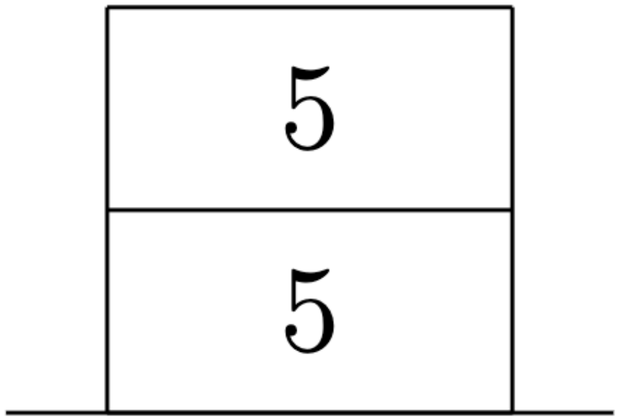
Risoluzione orphan jumps

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



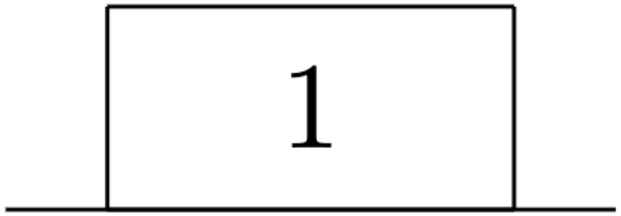
Risoluzione orphan jumps

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



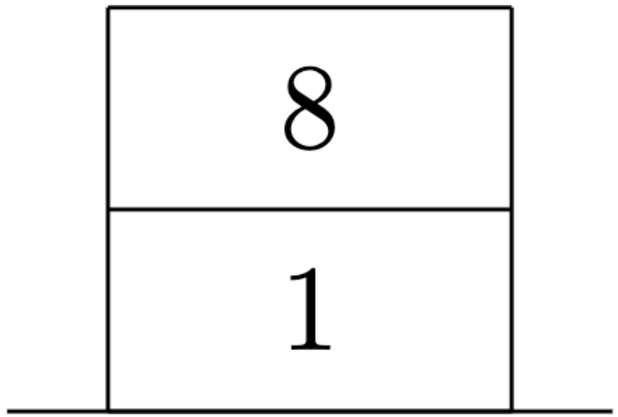
Risoluzione orphan jumps

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



Risoluzione orphan jumps

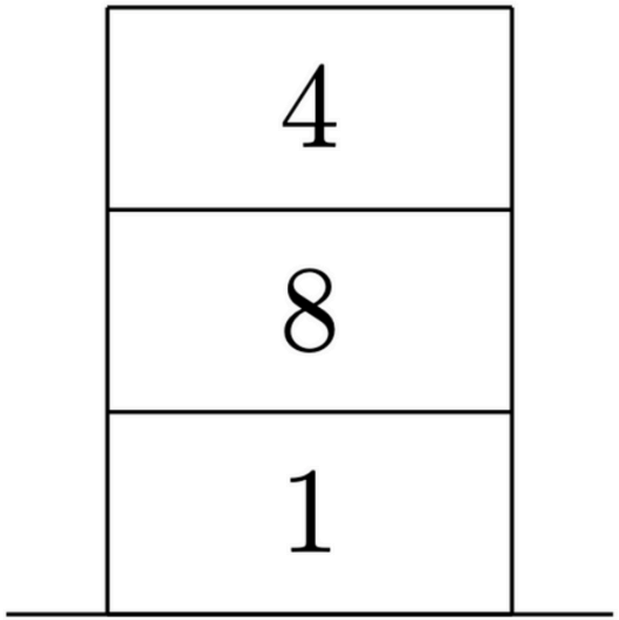
```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



Risoluzione orphan jumps

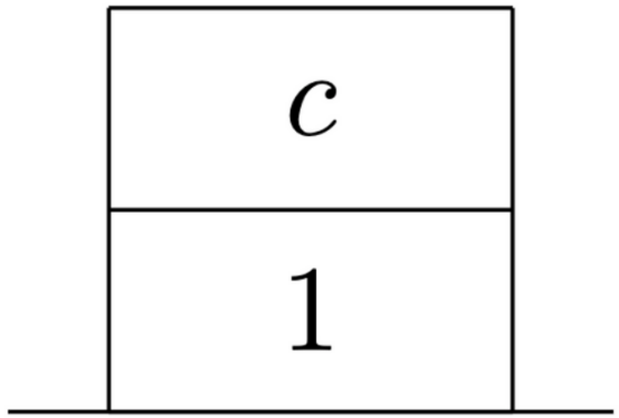
UNIVERSITÀ DI PARMA

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



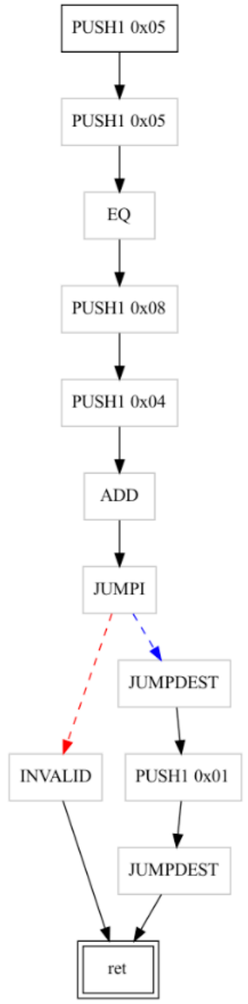
Risoluzione orphan jumps

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



Risoluzione orphan jumps

```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```



Valutazione sperimentale

UNIVERSITÀ DI PARMA

Benchmark su 5000 smart contract da *Etherscan*[1]

[1] https://etherscan.io/

Valutazione sperimentale

UNIVERSITÀ DI PARMA

Dimensione StackSet	Jump Risolte	Jump Unsound	Jump Irragg.	Maybe Unsound	Definitely Fake	Maybe Fake	% Jump Solved	Time (sec)
1	1728979	1	243285	315	851	48	99.9999%	3.44
2	1728688	6	223758	333	1125	60	99.9997%	4.24
4	1727825	20	196366	421	1950	84	99.9988%	7.38
8	1726589	15	154845	482	3089	186	99.9991%	14.99
16	1727152	18	129387	520	2321	367	99.9990%	25.97
32	1728251	0	113854	479	1491	137	100%	161.65

$$k = 1, h = 64$$

Valutazione sperimentale

UNIVERSITÀ DI PARMA

Benchmark su 5000 smart contract da *Etherscan*[1]

Confronto con *Ethersolve*[2] (94,61%)

[1] https://etherscan.io/

[2] "*Enhancing ethereum smart-contracts static analysis by computing a precise control-flow graph of ethereum bytecode*", Pasqua et al. [2023, *Journal of Systems and Software*]

Lavori futuri

UNIVERSITÀ DI PARMA

Ottimizzazione di EVMLiSA

Lavori futuri

UNIVERSITÀ DI PARMA

Ottimizzazione di EVMLiSA

Elaborazione di informazioni esterne

Lavori futuri

UNIVERSITÀ DI PARMA

Ottimizzazione di EVMLiSA

Elaborazione di informazioni esterne

Reentrancy & buffer overflow checker, gas estimator

Conclusioni

UNIVERSITÀ DI PARMA

Focus sulla creazione di un CFG sound

Conclusioni

UNIVERSITÀ DI PARMA

Focus sulla creazione di un CFG sound

Implementati diversi domini astratti

Conclusioni

UNIVERSITÀ DI PARMA

Focus sulla creazione di un CFG sound

Implementati diversi domini astratti

Ottenuti ottimi risultati

EVMLiSA

github.com/lisa-analyzer/evm-lisa

Sottomesso articolo al FTfJP 2024