NTT

NTT Information Sharing Platform Laboratories
NTT 情報流通プラットフォーム研究所

半透性仮想インターネットによるマルウェアの動的解析

○青木 一史† 川古谷 裕平† 岩村 誠†‡ 伊藤 光恭†

† NTT情報流通プラットフォーム研究所
‡ 早稲田大学

NTT

目次

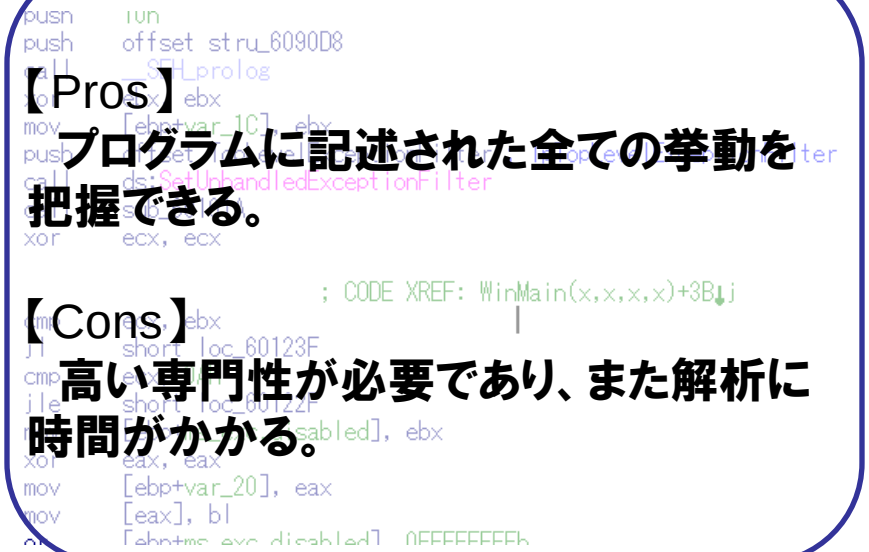
- 背景
- アプローチ
  - Botnet Watcher：半透性仮想インターネットにおける動的解析システム
  - 実行済みコード領域の分析
- CCC DataSet2009検体を用いた実験
- まとめ

NTT

背景

- ボットをはじめとするマルウェアは、日々新種・亜種が出現し、脅威をもたらしている。
  - マルウェアの挙動に応じた対策を講じるには、マルウェアの解析が必須。
  - 多数のマルウェアを短時間で解析するには、動的解析が効果的。

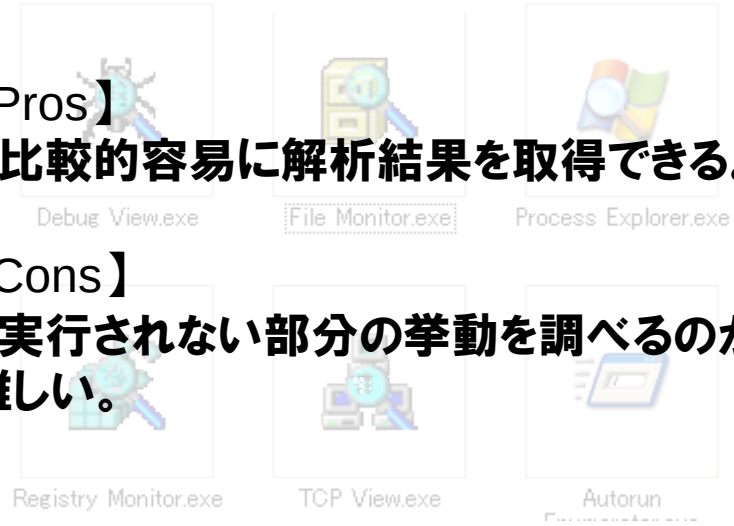
静的解析



【Pros】
プログラムに記述された全ての挙動を把握できる。

【Cons】
高い専門性が必要であり、また解析に時間がかかる。

動的解析



【Pros】
比較的容易に解析結果を取得できる。

【Cons】
実行されない部分の挙動を調べるのが難しい。

NTT

動的解析の課題

- 動的解析環境ごとの長所/短所
  - 閉環境（実インターネットから完全隔離）
    - 【Pros 】 安全に解析可能
    - 【Cons】 ボットやシーケンシャルマルウェアが実インターネットでどのような振る舞いを見せるのかを解析するのは困難
  - 開環境（実インターネットと接続可能）
    - 【Pros 】 ボットやシーケンシャルマルウェアの解析を可能
    - 【Cons】 適切なフィルタリングをしないと外部に攻撃をしてしまう可能性がある

ボットやシーケンシャルマルウェアを安全に解析する手法が必要

- 動的解析結果の評価における課題
  - 動的解析では、解析時に実行された領域の解析結果しか得られない
    - 解析時に実行されるコード量が多いほど、その検体の挙動を多く把握できるようになると考えられる。

解析環境の違いにより、実行されるコード量がどの程度変化するのか不明

NTT

アプローチ

【課題①】

ボットやシーケンシャルマルウェアを安全に解析する手法の確立

→ Botnet Watcher：
半透性仮想インターネットによる動的解析

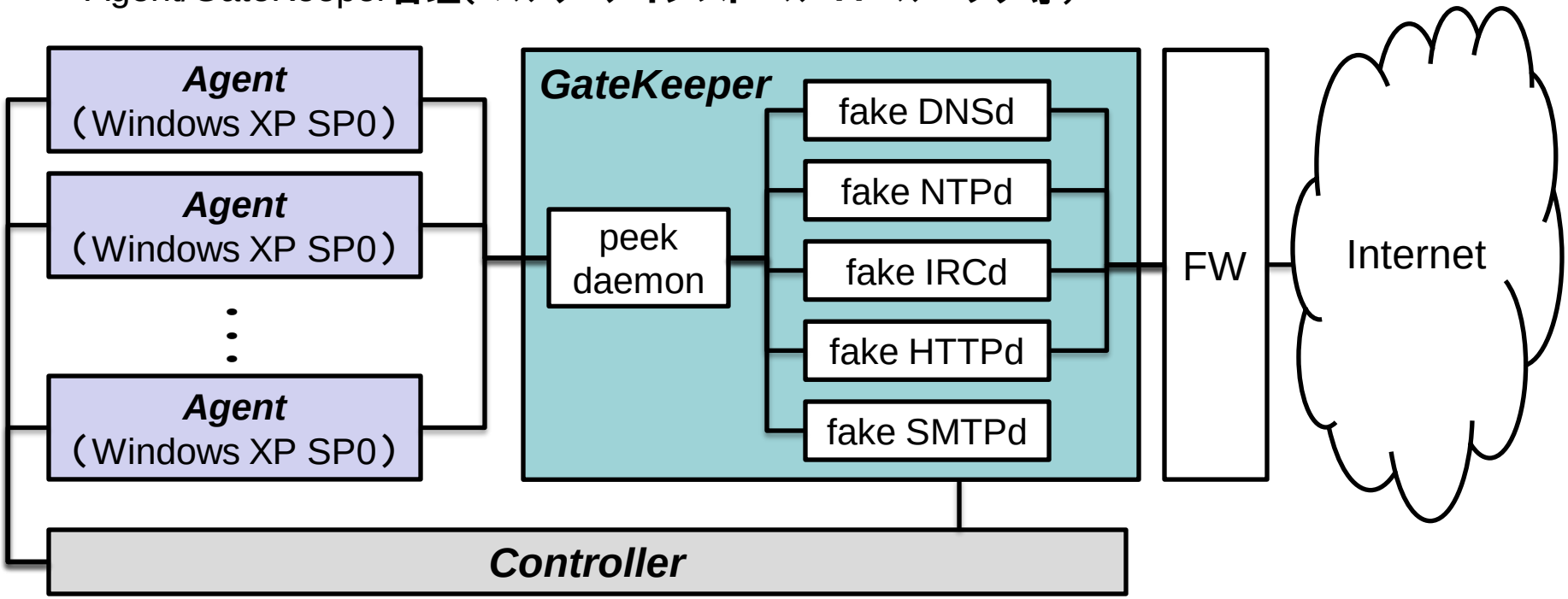
【課題②】

解析環境の違いと実行済みコード量の分析

→ 閉環境・開環境における動的解析時に
実行されたコード領域の比較評価

Botnet Watcher：半透性仮想インターネットによる動的解析

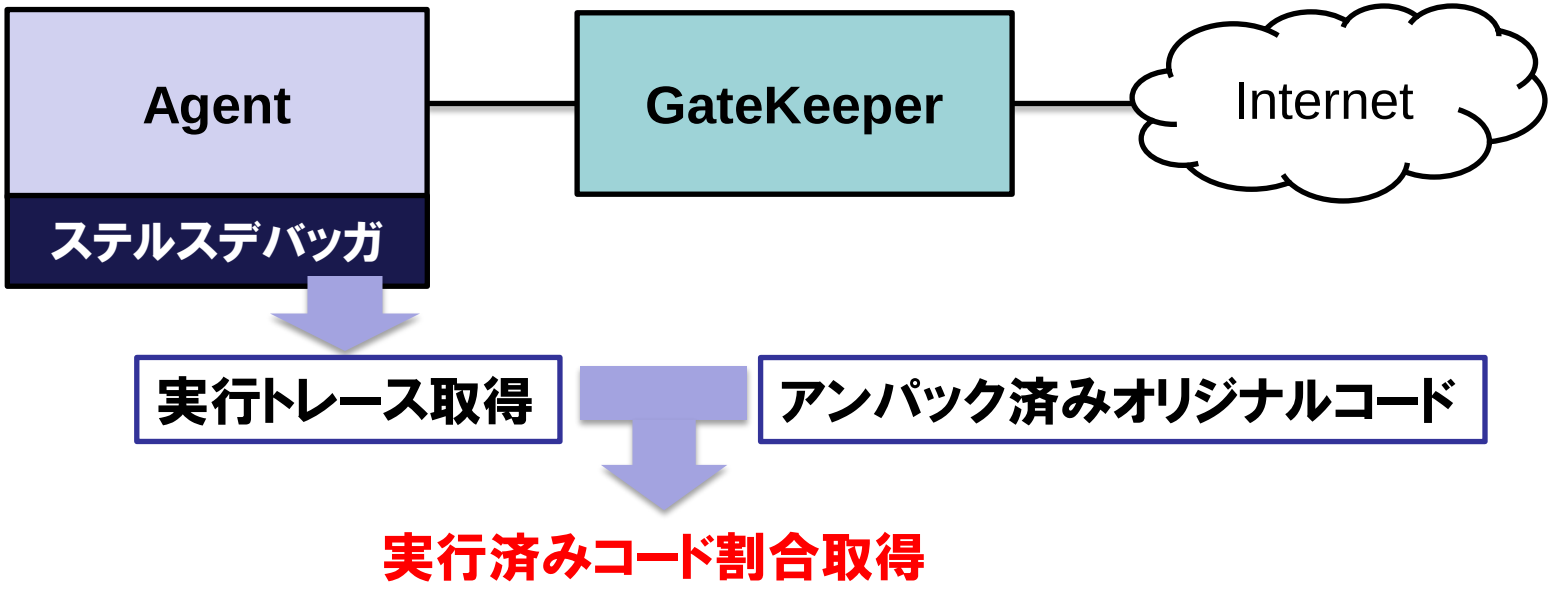
- ***Agent***：
  - マルウェア実行環境（Anti-VM対策済み）
- ***GateKeeper***：
  - 半透性仮想インターネット環境
    - 半透性仮想インターネット：ペイロードからプロトコルを識別し、マルウェアの通信を実インターネットに通過させるか、擬似的な応答を返す仮想インターネットで対応するかを識別する。
- ***Controller***：
  - Agent/GateKeeper管理（マルウェアインストール・ロールバック等）



NTT

実行済みコード領域の分析

- 閉/開の動的解析環境でマルウェアを動作させる。
  - 解析中に、ステルスデバッガ*1でマルウェアの実行トレースを取得。
    - Anti-Debug対策
    - 高速なトレースを実現
- 取得した実行トレース結果と、アンパック済みのオリジナルコード*2から、実行済みコード割合を取得。



*1…川古谷ら:ステルスデバッガを利用したマルウェア解析手法の提案（MWS2008）
*2…岩村ら:コンパイラ出力コードモデルの尤度に基づくアンパッキング手法（MWS2008）

NTT

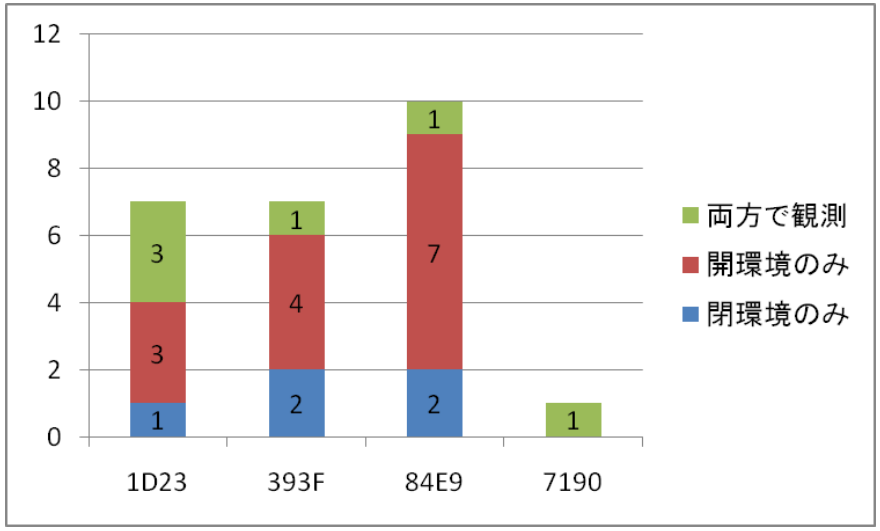
実験環境

- 閉/開環境における動的解析
  - 解析対象検体
    - CCC DataSet 2009検体 （10検体）
  - 解析時間
    - 1検体につき3分間動作
  - 解析環境
    - 閉環境：Matrix Daeomns
      - 隔離環境における動的解析システム
        - » TCP・・・セッション確立まで応答
        - » UDP・・・NTP/DNSに対して擬似応答
    - 開環境：Botnet Watcher
      - 半透性仮想インターネットによる動的解析システム
        - » TCP・・・HTTP/IRC/SMTPをプロトコル判別し、実インターネットor仮想インターネットが応答
        - » UDP・・・NTP/DNSに対して、実インターネットと同様の値で応答

NTT

動的解析結果(接続先IP数)

hash	TCP						UDP		
	IRC		HTTP		UNKNOWN		DNS		
	閉	開	閉	開	閉	開	閉	開	
1D23(W32.Mancsyn)			3	34		172	1	1	Downloader
393F(Trojan.Horse)			3	5		7,451	1	1	
84E9(Downloader)			3	5		4,566	1	1	
7190(W32.Spybot.Worm)	1	1				3,395	1	1	IRCボット



問い合わせFQDN数

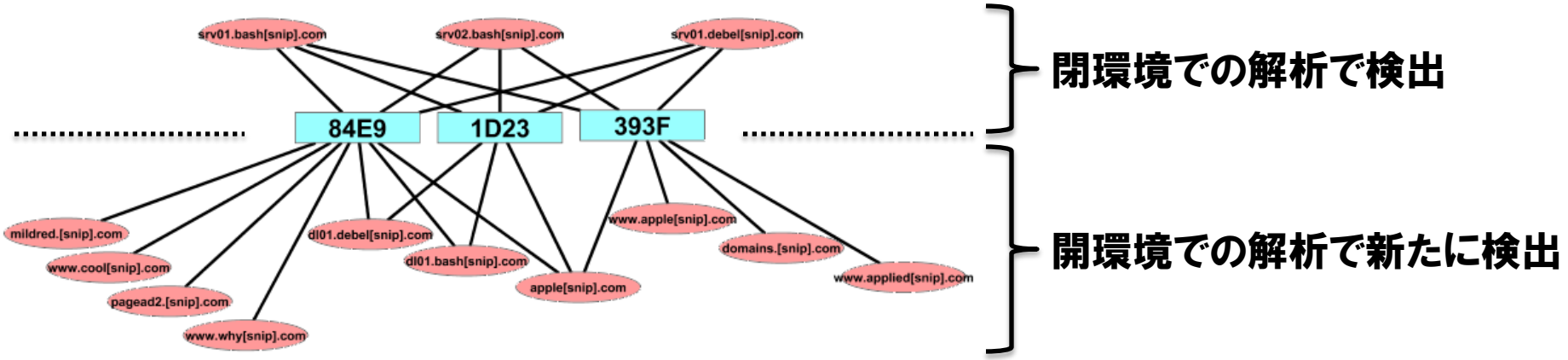
閉環境時には見られなかった外部へのスキャンを開環境での解析で観測

ボットやシーケンシャルマルウェアの活動観測を実現

問い合わせFQDN

- 閉環境での解析では、3つの検体が3種類の同一FQDNの名前解決を実施。
- 開環境で解析したときに各検体から追加で得られた接続先は必ずしも一致していない。

→ シーケンシャルマルウェアの場合、実際の脅威を分析するためには開環境での解析が効果的



検体と問い合わせFQDNの関係

動的解析結果(接続先IP数)

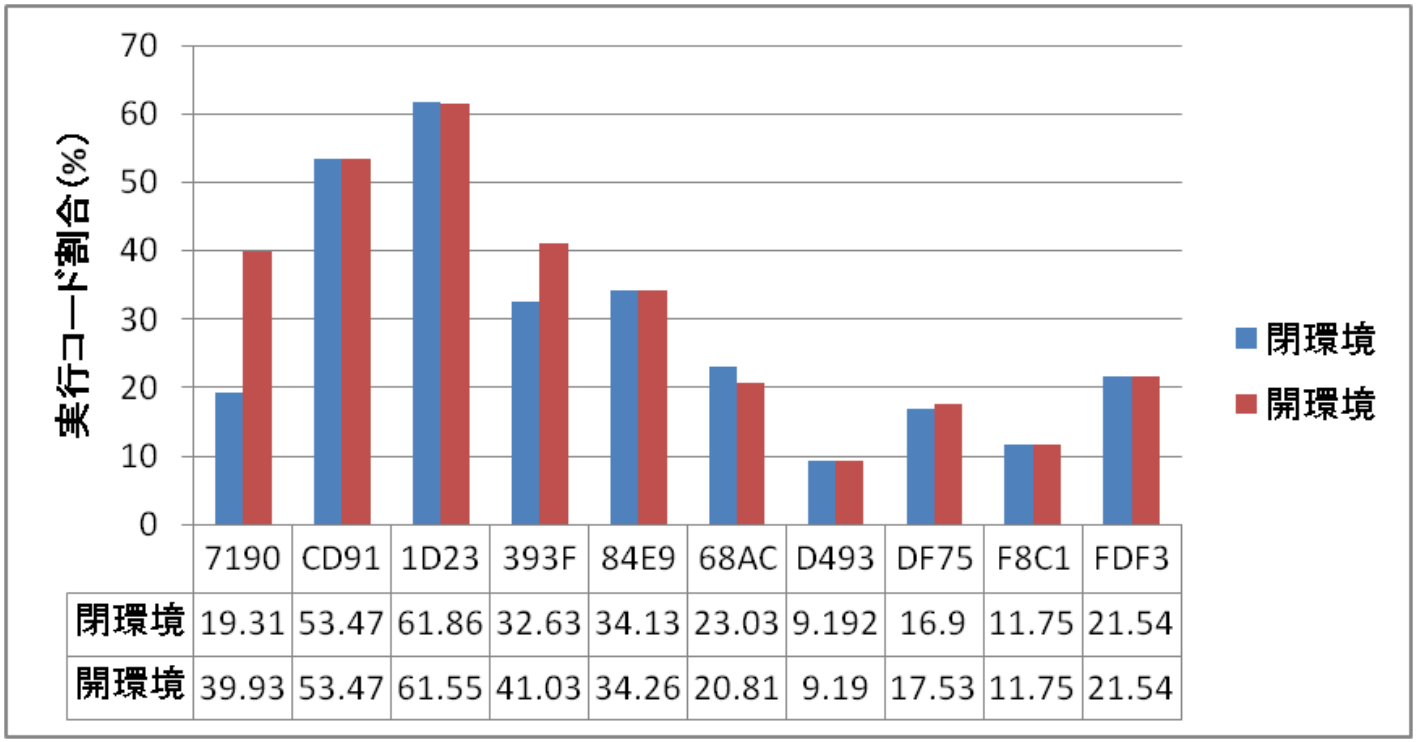
hash	TCP						UDP					
	IRC		HTTP		UNK		DNS		NTP		UNK	
	閉	開	閉	開	閉	開	閉	開	閉	開	閉	開
68AC(Trojan.Peacomm)			1	1			1	1	1	1	145	114
CD91(W32.Spybot.Worm)			44	73	93	87					6	
D493(W32.Virut.B)	1						1	1				
DF75(Trojan.Hose)			12	11								
F8C1(Spyware.ISearch)			2	1			1	1				
FDF3(Trojan.Horse)			1	1			1	1				

開環境では、IRCの接続先FQDNの名前解決ができなかったため。
(2009/8/19**時点**)

Botnet Watcher**で許可していない通信のため、閉/開環境で大きな差が出ない。**

安全性と解析精度のトレードオフについて、まだ課題がある

実行済みコード領域の実験結果

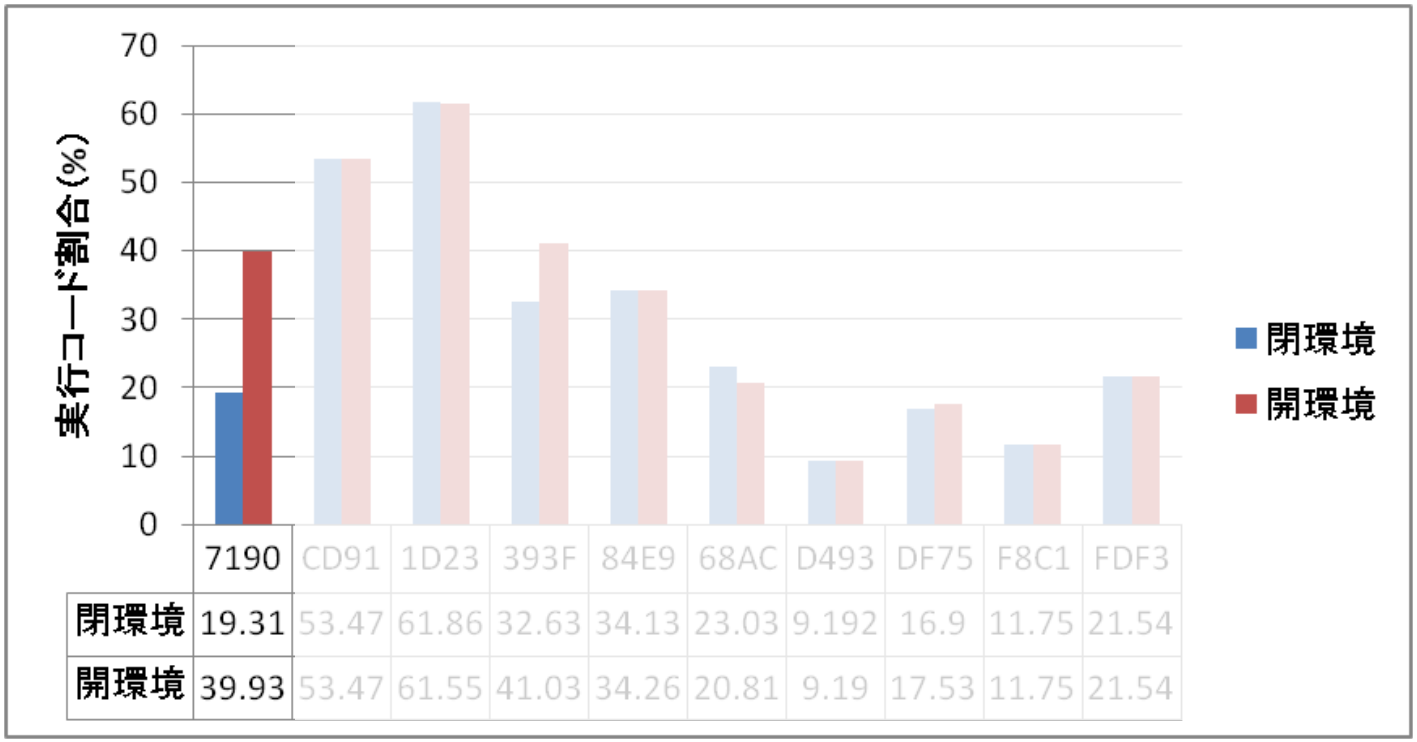


	7190	CD91	1D23	393F	84E9	68AC	D493	DF75	F8C1	FDF3
閉環境	19.31	53.47	61.86	32.63	34.13	23.03	9.192	16.9	11.75	21.54
開環境	39.93	53.47	61.55	41.03	34.26	20.81	9.19	17.53	11.75	21.54

- **閉環境でも開環境でも、動的解析中に全てのコードが実行されていない。**
  - CCC DataSet2009**検体の場合、**10～60%**程度のコードが実行されている**

NTT

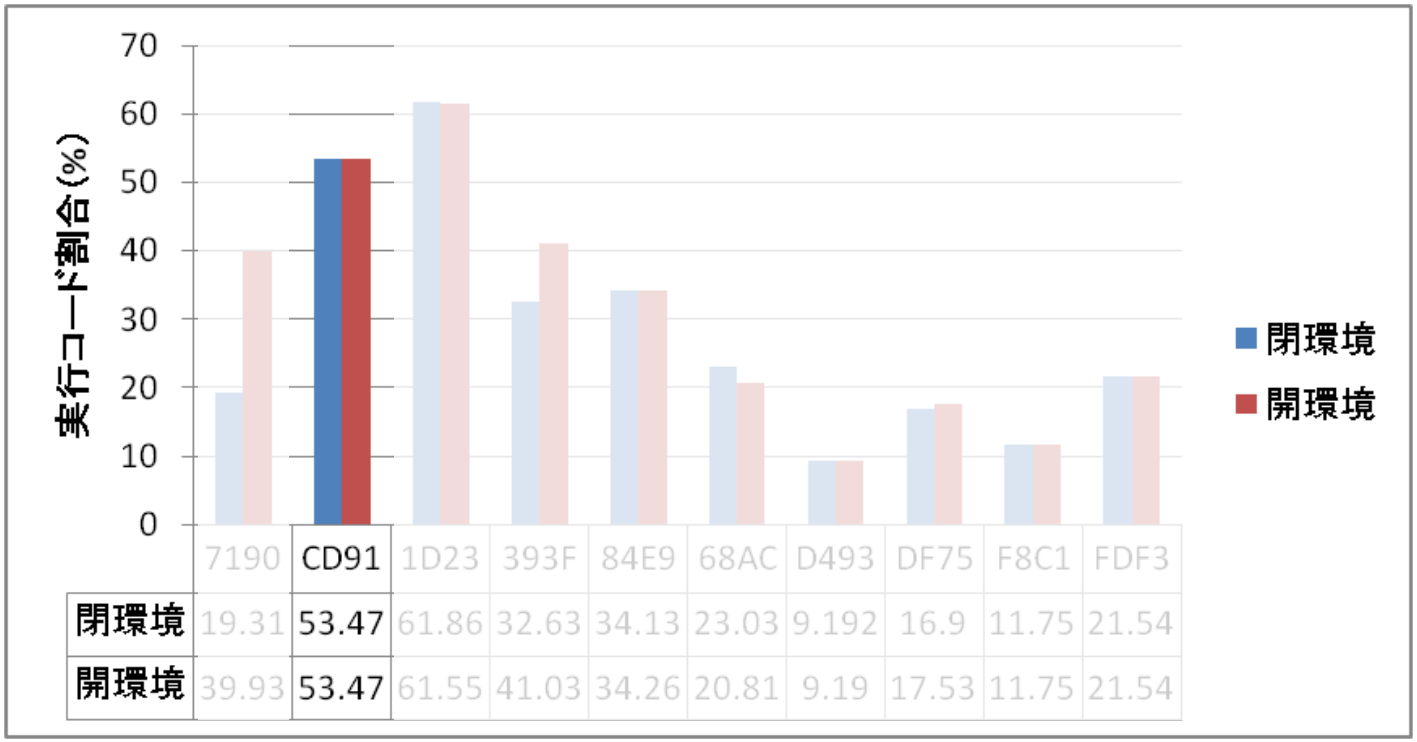
実行済みコード領域の実験結果



	7190	CD91	1D23	393F	84E9	68AC	D493	DF75	F8C1	FDF3
閉環境	19.31	53.47	61.86	32.63	34.13	23.03	9.192	16.9	11.75	21.54
開環境	39.93	53.47	61.55	41.03	34.26	20.81	9.19	17.53	11.75	21.54

- ボットのように、外部からの指令に応じて挙動が変化するマルウェアの場合は、開環境の方が実行されるコード量が増加。

実行済みコード領域の実験結果



	7190	CD91	1D23	393F	84E9	68AC	D493	DF75	F8C1	FDF3
閉環境	19.31	53.47	61.86	32.63	34.13	23.03	9.192	16.9	11.75	21.54
開環境	39.93	53.47	61.55	41.03	34.26	20.81	9.19	17.53	11.75	21.54

- ワームのように、外部からの指令を特に必要としない検体は、閉環境でも開環境でも、実行されるコード量に変化はない。

NTT

まとめ

- 開環境での動的解析手法として、Botnet Watcherを提案
  - CCC DataSet 2009検体を使った実験より、ボットやシーケンシャルマルウェアの実インターネットにおける挙動を分析した。
  - 安全性と解析精度を両立させるために、外部との通信の許可方式については更なる検討が必要である。

- 動的解析時に実行されるコード領域の差異を追跡
  - ボットのような、外部からの指令に起因して挙動が変化するような検体の解析では、開環境がより多くのコード領域が実行される。
  - 単純なワームのように、外部からの通信を必要としない検体については、解析環境の違いで、実行されるコード領域は変化しない。