SiamHAN: IPv6 Address Correlation Attacks on TLS Encrypted Traffic via Siamese Heterogeneous Graph Attention Network

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

usenix
THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

Tianyu Cui, Gaopeng Gou, Gang Xiong, Zhen Li, Mingxin Cui, and Chang Liu

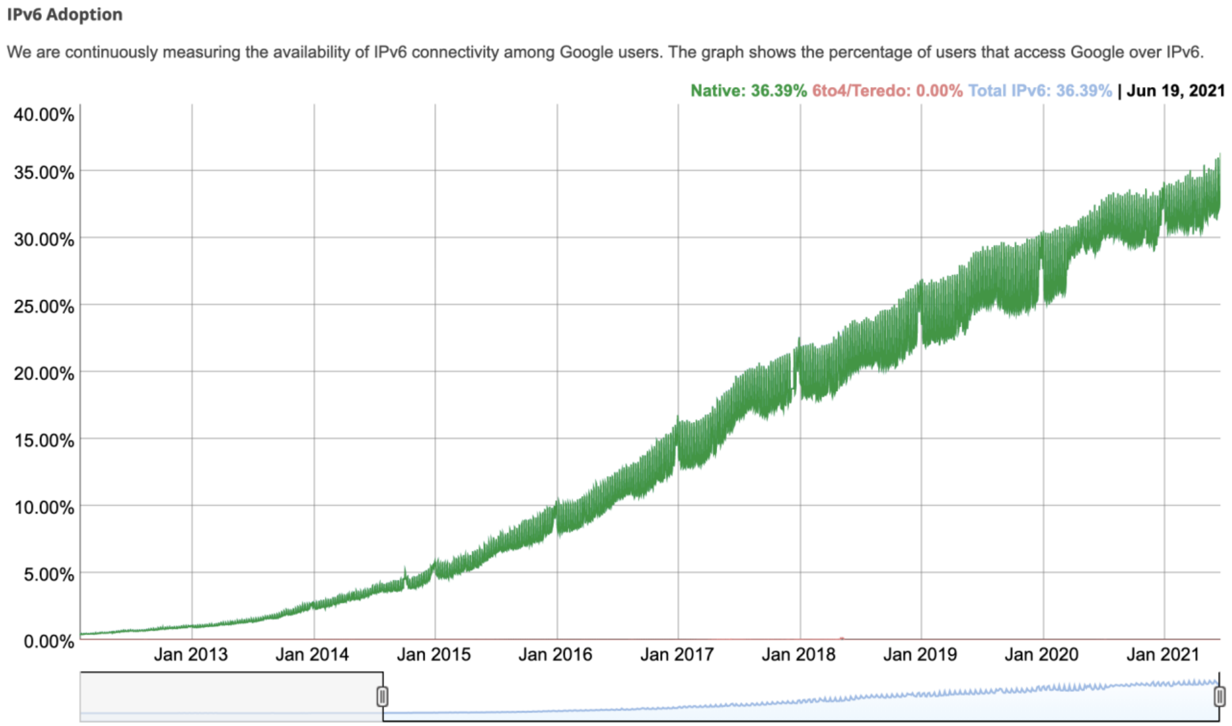
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences

IPv6 Networks

The Growth of IPv6 Networks

- An increasing number of network providers expediting the deployment of IPv6
- One-third of Internet users can now access online services through IPv6

- **Increased focus on IPv6 security and privacy issues**



Don't Forget to Lock the Back Door! A Characterization of IPv6 Network Security Policy

Towards A User-Level Understanding of IPv6 Behavior

Who Knocks at the IPv6 Door? Detecting IPv6 Scanning

Privacy is Not an Option: Attacking the IPv6 Privacy Extension

Flaw Label: Exploiting IPv6 Flow Label

Security and Privacy Considerations for IPv6 Address Generation Mechanisms

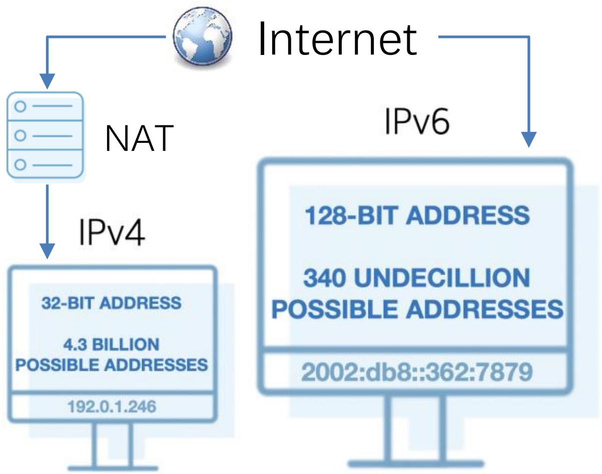
IPv6 User Activity Correlation

User Activity Correlation

- Leveraging traffic meta-information to identify and track users
- Could work even on traffic encrypted by Transport Layer Security (TLS)

Work on IPv6

- Unlike IPv4 - rare deployment of Network Address Translation (NAT)
- An IPv6 address usually corresponds to one single user
- **Serious individual-level privacy threat!**



Limitation

Address-based Correlation

- Associating an IPv6 address with a user’s activity
- Weak configuration - a CONSTANT interface identifier:

  2001:db8::face:b00c:0:a7 ----> 2001:db8::face:b00c:0:a7
- Mitigation - temporary addresses (RFC 4941):

  2001:db8::7c61:2880:3148:36e1 ----> 2001:db8::6efb:720a:8321:92dc

  Dynamic changing and pseudorandom

Traffic Characteristic Correlation

- Associating traffic with a user’s activity
- Analyzing the patterns in the encrypted traffic
- Limitation - closed-world dataset:

  Can only correlate the traffic of a selected subset of users (only known users)

IPv6 Address Correlation Attack

Challenge

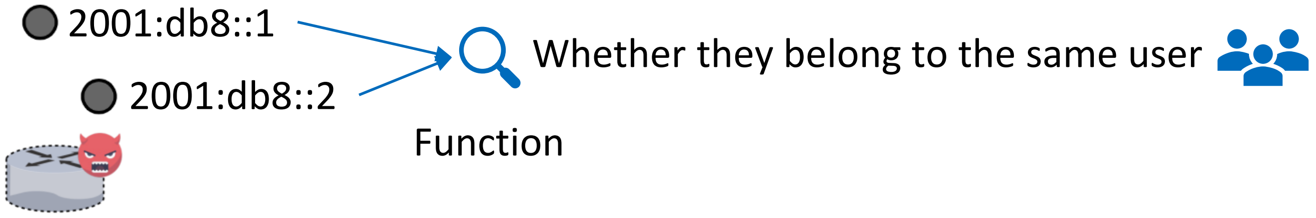
Frequently changing client addresses
Widespread payload encryption with TLS

Making address-to-user correlation unreliable

Our Attack

Learning a correlation function from TLS encrypted traffic

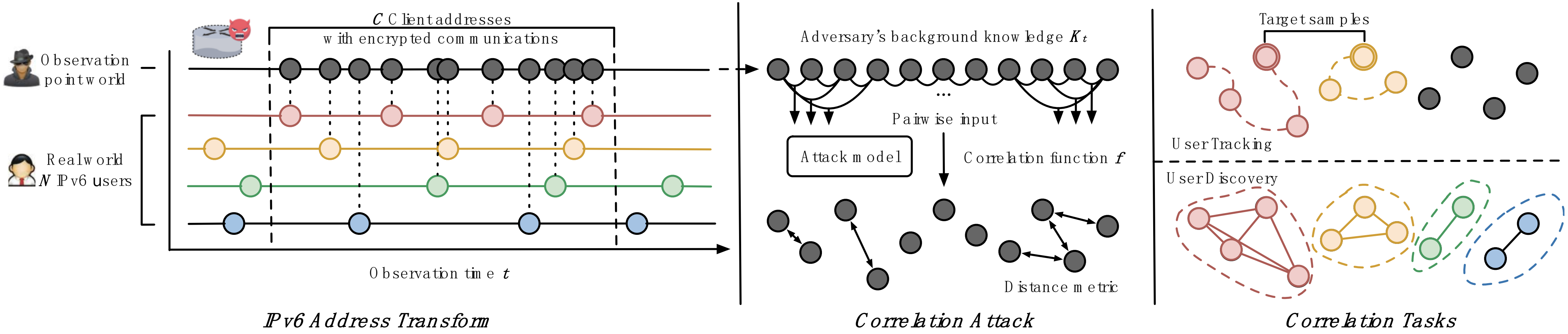
Two arbitrary addresses



2-step attack:

- Construct **knowledge graph** for each client address
- Capture the relationship between each two addresses with an **attack model**

Threat Model



Attack Scenario

- An observation point for wiretapping
- **Adversary's background knowledge K_t** :
  - Encrypted communication behavior of all IPv6 addresses during the wiretapping time *t*
- **Correlation function *f*** :
  - Judging the relationship of a pair of addresses
  - Learned by an attack model - distance metric with a threshold η

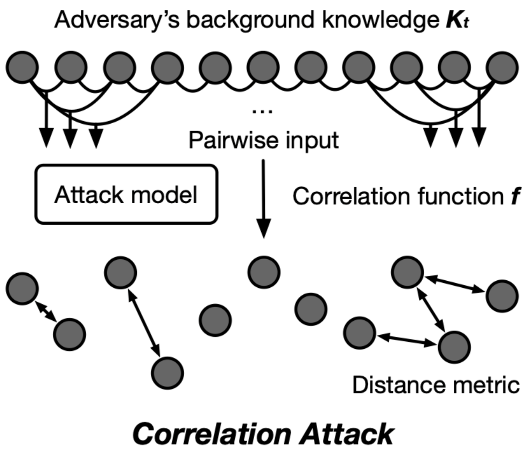
Adversary Ground Truth

Labeling Trick

- Leaked persistent cookie
  - A few users use the changing addresses and access some websites without HTTPS deployment
  - The TLS connections of these addresses could be labeled

- Simulating and generating user data by adversary’s own clients

- The adversary could perform large-scale correlation attacks on the wild TLS traffic without plaintext once obtaining the model



SiamHAN

Knowledge Graph

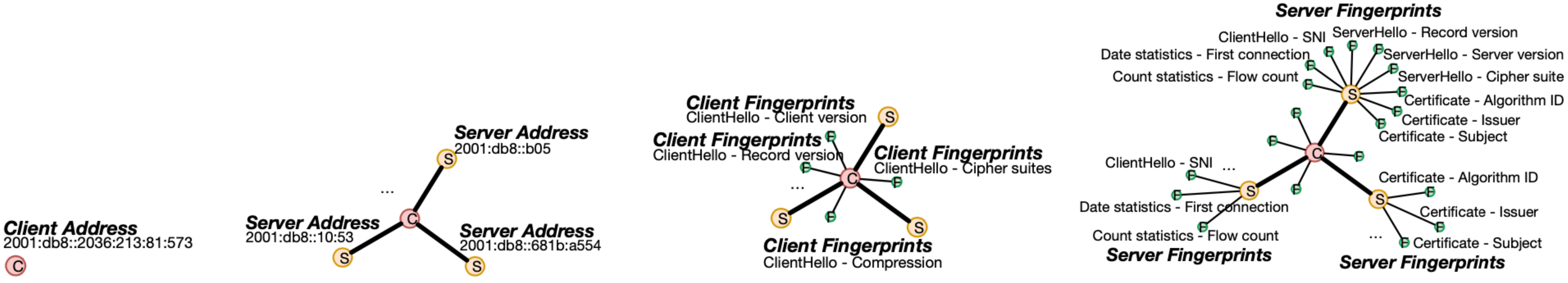
Heterogeneous graph - multi-type nodes and neighbor relationships

Node and Node Attribute

- **Client node C**
  - The 32-digit hexadecimal IPv6 client address
  - Each graph only have one
- **Server node S**
  - The 32-digit hexadecimal IPv6 server address who have established TLS communications with the client
- **Fingerprint node F**
  - Field values of the ClientHello, ServerHello, Certificate messages, and statistical characteristics
  - Client fingerprints and server fingerprints

Node Type	Source	Label	Node Attribute
Client node	IPv6 header	C	Client address
Server node	IPv6 header	S	Server address
Client fingerprint	ClientHello	F_1	Record version
		F_2	Client version
		F_3	Cipher suites
		F_4	Compression
Server fingerprint	ClientHello	F_5	SNI
	ServerHello	F_6	Record version
		F_7	Server version
		F_8	Cipher suite
	Certificate	F_9	Algorithm ID
		F_{10}	Issuer
		F_{11}	Subject
	Date statistics	F_{12}	First connection
	Count statistics	F_{13}	Flow count

SiamHAN



(a) Node C (b) Nodes S and SCS meta-paths (c) Nodes F and FCF meta-paths (d) Nodes F and FSF meta-paths

Knowledge Graph

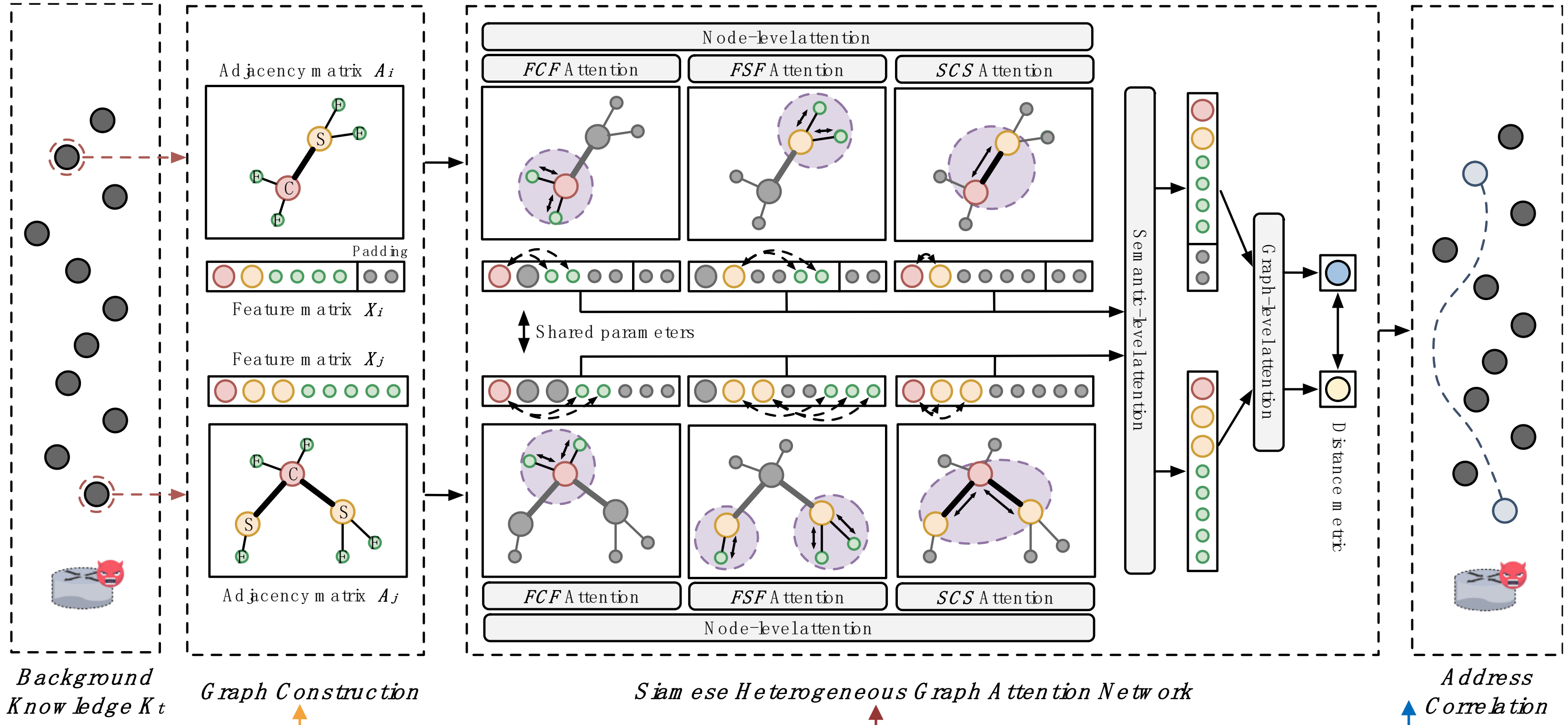
Neighbor Relationship

- **SCS meta-path -** Connecting C and S
    - The TLS communication activities between the client and multiple servers
- **FCF meta-path -** Connecting C and client fingerprint F
    - The browser parameters that may be used behind the client
- **FSF meta-path -** Connecting S and server fingerprint F
    - The service characteristics behind each server

SiamHAN

Model Architecture

Input the knowledge graphs of two arbitrary client addresses



Extract an adjacency matrix *A* and feature matrix *X* for each graph

Multi-level attention: match the similar features between the two heterogeneous graphs to learn their graph embeddings

Measure the distance of the embeddings to judge the correlation relationship

SiamHAN

Model Architecture

Node-level attention

Locally match similar nodes in each single meta-path-based semantic

- Learning the importance of meta-path-based neighbors and aggregating them to get the semantic-specific node embeddings (SCS FCF FSF)

Semantic-level attention

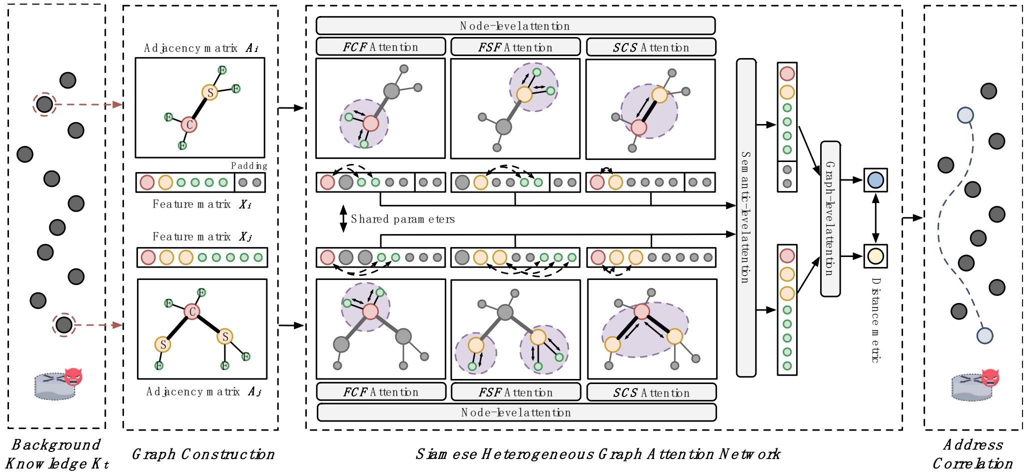
Semantic aggregation

- learning the importance of three types of semantic-specific embeddings for each node and fusing them as comprehensive node embeddings

Graph-level attention

Globally match similar nodes

- aggregating the comprehensive embeddings of all nodes in the knowledge graph to get the graph embedding



Metric learning with Siamese Network

- Measuring the **distance *D*** between the two graphs and judging the correlation **relationship *R*** through a **threshold η**

$$D = ||Z_1 - Z_2||_2,$$

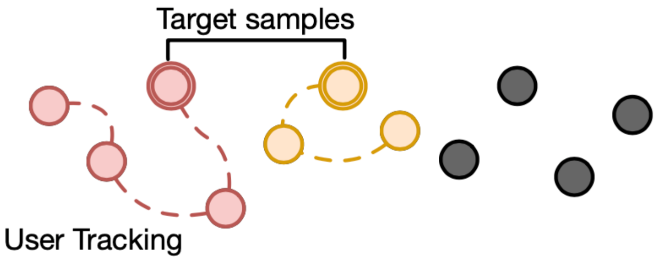
$$R = \begin{cases} 1 & D < \eta \\ 0 & D \geq \eta \end{cases},$$

- Contrastive loss function

$$L = Y \cdot D^2 + (1 - Y)\{\max(0, m - D)\}^2$$

SiamHAN - User Tracking

User Tracking Task



Searching all addresses correlated to the address sample of target users

- Target users' one client address activity is known
- The adversary could compute the relationship between **each target address** and **each test address** during the observation

Algorithm 1 The tracking algorithm applied by SiamHAN

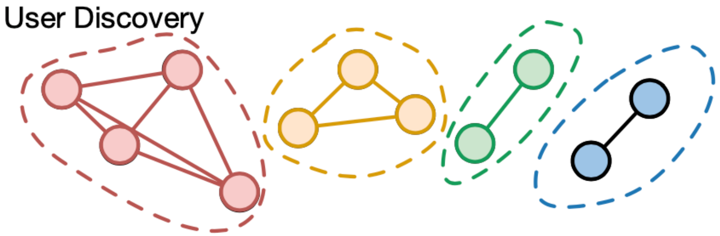
Require: Pre-trained SiamHAN ρ; Tracking candidate set S; Test address set T; Background knowledge κ_t.

Ensure: Address sets T_{S_i} link to the same user with each S_i

```
for S_i in tracking candidate set S, where i ≤ |S| do
    Initialize target address set T_{S_i} = {}
    for T_j in test address set T, where j ≤ |T| do
        Build pairwise knowledge graphs for ⟨S_i, T_j⟩
        Test relationship R of ⟨S_i, T_j⟩ using pre-trained ρ
    end for
    Append T_j in address set T_{S_i} if relationship R = 1
end for
return T_{S_i} for each S_i
```

SiamHAN - User Discovery

User Discovery Task



Calculating the correlation between every two addresses to acquire address clusters

- The number of users in traffic is unknown
- The adversary could use a recursion algorithm to determine the unique users

Algorithm 2 The discovery algorithm applied by SIAMHAN

Require: Pre-trained SIAMHAN ρ; Discovery candidate set S; Background knowledge κ_t; Task threshold η.

Ensure: User groups G under the discovery candidate set S

```
Build knowledge graphs for each S_i
Initialize user group set G = {G_1}
Initialize S_1 into the first user group G_1
for S_i in discovery candidate set S, where 1 < i ≤ |S| do
    for G_k in user group set G do
        for Address S_j in group G_k, where j ≤ |G_k| do
            Calculate distance D for ⟨S_i, S_j⟩ using ρ
        end for
        Calculate average distance D̄_k for S_i to G_k
    end for
    if All group average distance D̄_k > η then
        Initialize a new user group G_{|G|+1} into G
        Initialize S_i into the new user group G_{|G|+1}
    else
        Classify S_i into G_k with the minimum D̄_k
    end if
end for
return User group set G
```

Evaluation - Dataset

Dataset Composition

- Passively collected on China Science and Technology Network (CSTNET) from March to July 2018
- Labeling - persistent cookie
- 1.7k IPv6 users with TLS traffic

AS Name	%Hits	Device OS	%Hits	SNI	%Hits	TLS Field	%Hits
CSTNET	78.6%	Windows	63.7%	*.google.com	17.9%	Record version	93.1%/ 93.9%
China Unicom	10.1%	Android	23.7%	*.adobe.com	11.6%	Client version	93.1%
CNGI-CERNET2	4.0%	iOS	6.2%	*.microsoft.com	11.2%	Server version	93.9%
CERNET	2.4%	Linux	5.0%	*.gstatic.com	4.8%	Cipher suites	93.1%/ 93.9%
Reliance Jio	1.6%	Mac OS X	1.3%	*.macromedia.com	3.3%	Compression	93.1%
Cloudflare	0.8%	BlackBerry	0.1%	*.cloudflare.com	2.4%	SNI	93.1%
PKU6-CERNET2	0.5%	Chrome OS	0.1%	*.2mdn.net	1.9%	Algorithm ID	78.4%
TSINGHUA6	0.5%	Symbian OS	0.1%	*.xboxlive.com	1.6%	Issuer	78.4%
ZSU6-CERNET	0.4%	Firefox OS	0.1%	*.xhcdn.com	1.2%	Subject	78.4%

Time-based Data Split

- Realistic setting from an adversary
    - First 3-month data for training
    - The 4th month's data for validation
    - The 5th month's data for test

Entity	Training	Validation	Test
User	1.0k	0.2k	0.5k
Sample Pair	1.2M	0.1M	0.2M
Knowledge	3 months	1 month	1 month

Evaluation - Baselines and Metrics

Baselines

User IP Profiling [1] - building user profiles through all the destination IPs of the client address and using a Bayesian classifier

User SNI Profiling [2] - using the SNIs in all the TLS ClientHello messages from the client as a user profiles and using a Bayesian classifier

Client Fingerprinting [3] - extracting the fields of the TLS ClientHello message as the user's client fingerprints and using a Random Forest classifier

Deepcorr [4] - using the flow sequence characteristics to achieve correlation tasks with a deep learning model

Metrics

- True Positive Rate (TPR)
- False Positive Rate (FPR)
- Area Under Curve (AUC)
- Accuracy
    - Tracking Accuracy (TA)
    - Discovery Accuracy (DA)

[1] Marek Kumpost and Vashek Matyas. User profiling and re-identification: Case of university-wide network analysis. In *TrustBus*, pages 1–10, 2009
[2] Roberto Gonzalez, Claudio Soriente, and Nikolaos Laoutaris. User profiling in the time of HTTPS. In *IMC*, pages 373–379, 2016
[3] Blake Anderson and David A. McGrew. OS fingerprinting: New techniques and a study of information gain and obfuscation. In CNS, pages 1–9, 2017
[4] Milad Nasr, Alireza Bahramali, and Amir Houmansadr. DeepCorr: Strong flow correlation attacks on tor using deep learning. In CCS, pages 1962–1976, 2018

Evaluation - Analysis of Hierarchical Attention

Analysis of Node-level Attention

High attention value of node *C*

- Matching the same constant IID in the two client addresses

High attention value of node *S* or *F*

- Matching the common server address or fingerprints

Analysis of Semantic-level Attention

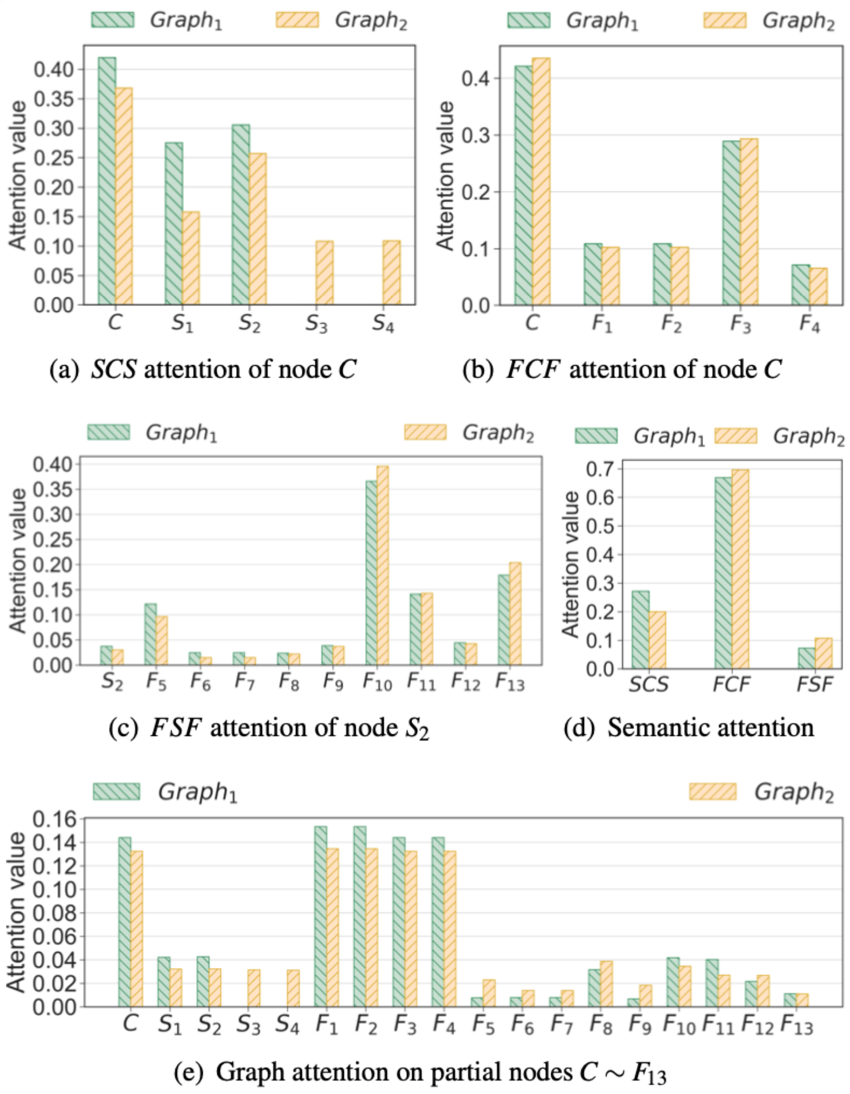
High attention value of *FCF* meta-path

- Learning user browser parameters is more important

Analysis of Graph-level Attention

F_1~ F_4 - Client Fingerprints; F_5~ F_{13} - Server Fingerprints

- Taking more attention to the client meta-information than the server meta-information



(a) *SCS* attention of node *C*

(b) *FCF* attention of node *C*

(c) *FSF* attention of node S_2

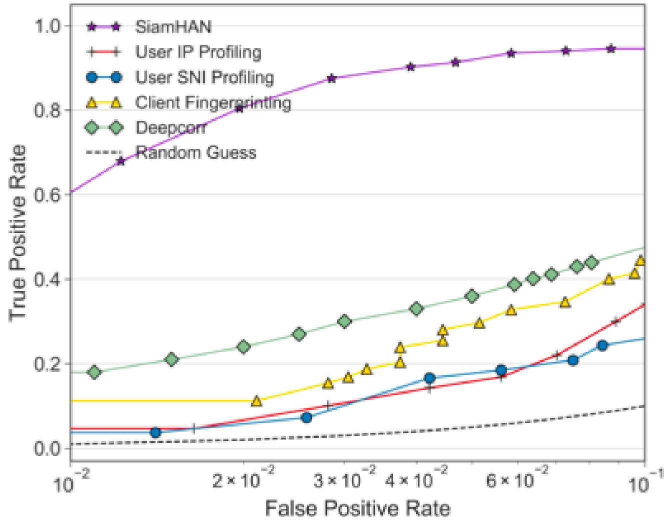
(d) Semantic attention

(e) Graph attention on partial nodes $C \sim F_{13}$

Evaluation - Address Correlation

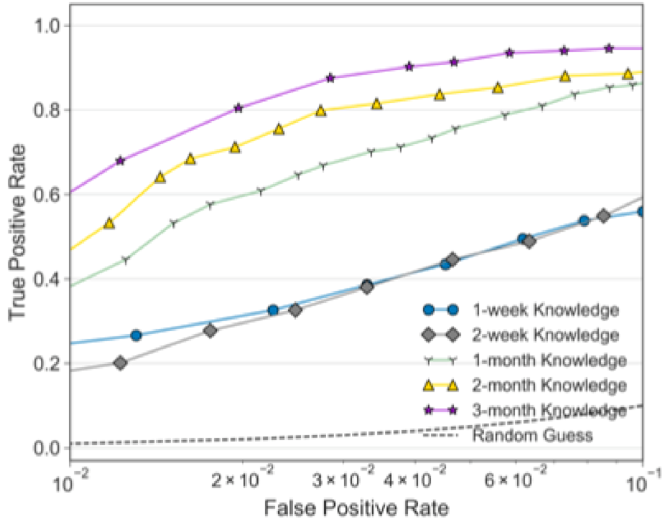
Correlation Performance

For a target FPR = 4×10^{-2}, while SiamHAN achieves a TPR of 0.90, all baselines provide TPRs less than 0.40



Adversary's Background Knowledge

SiamHAN's performance is positively correlated with the volume of the adversary's background knowledge on the training set

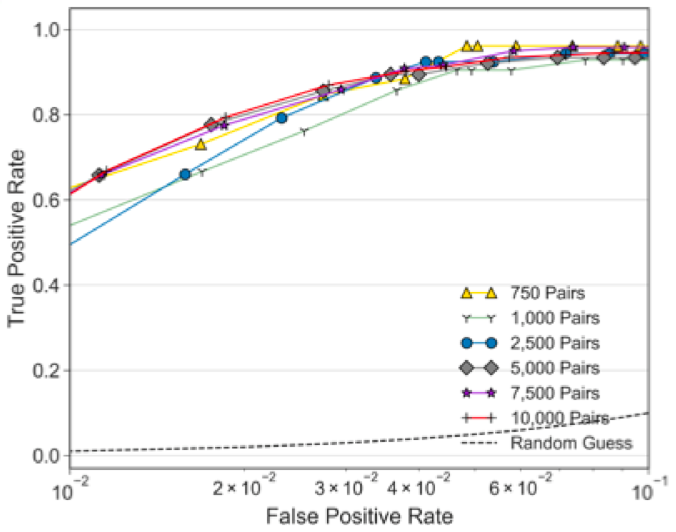


Evaluation - Address Correlation

Robustness of Test Users

Evaluating on the different sizes of the test dataset

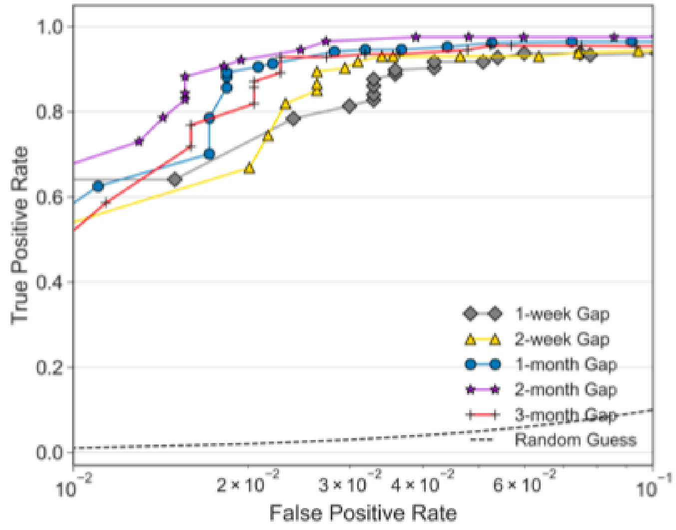
The correlation performance is consistent for different numbers of addresses being correlated



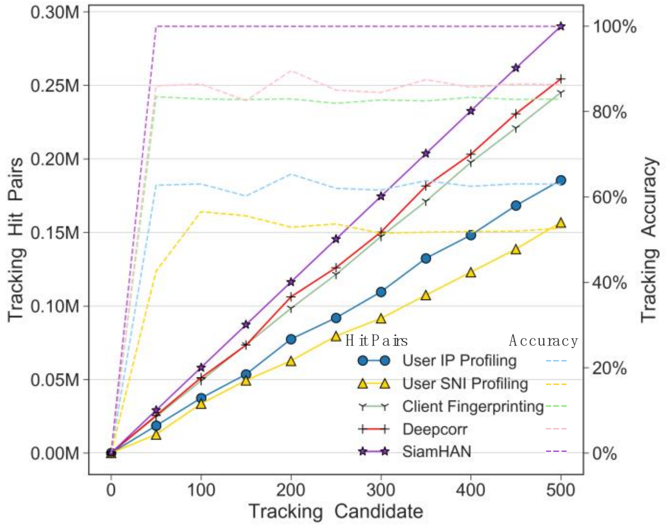
Timeliness

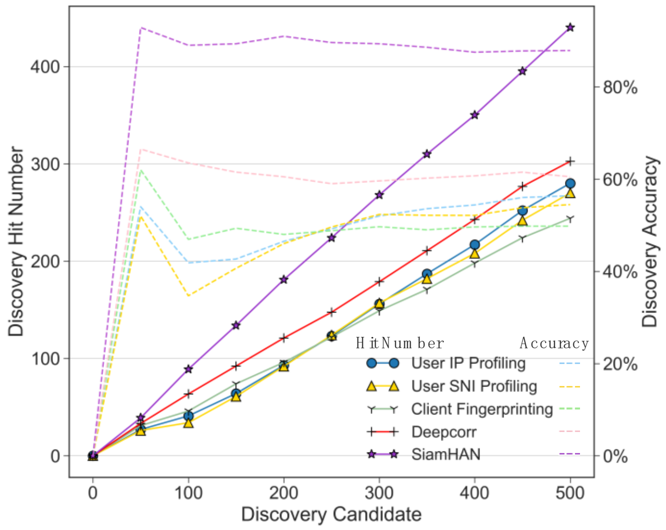
Evaluating on the different time gaps between training and test

For a target FPR = 10−1, under all time gaps, SiamHAN provides TPRs more than 0.95



Evaluation - User Tracking and User Discovery





SiamHAN outperforms existing correlation techniques with 99% and 88% accuracy compared to 85% and 60% accuracy of the best baseline on the user tracking and user discovery task

SiamHAN could achieve 1.10~1.19 and 1.40 ~ 1.54 times more hit than Deepcorr

Evaluation - Countermeasures

Traffic Obfuscation

- C-Random - random forged client address
- CF-Random - random browser parameters
- CF-background - adding background traffic of different browsers
- SF-background - adding background traffic of different online services
- The combination of all four methods - knowledge barrier

Obfuscation Method	Address Correlation	User Tracking	User Discovery
C-Random	0.855	0.905	0.808
CF-Random	0.878	0.897	0.810
CF-Background	0.871	0.922	0.823
SF-Background	0.893	0.910	0.830
Combination	0.705	0.769	0.643

Attack Chance Reduction

- Escape - **Tor system, proxy**
- Meta-information protection - **encrypted VPN**
- Address-user relation protection - **NAT**

Conclusion

- We explore the implementation of user activity correlation on IPv6 networks.
- We propose IPv6 address correlation attacks, which leverage an attack model SiamHAN to learn the correlation relationship between two arbitrary IPv6 addresses based on the background knowledge of TLS traffic.
- We hope that our work demonstrates the serious threat of IPv6 address correlation attacks and calls for effective countermeasures deployed by the IPv6 community.

THANK YOU FOR LISTENING

Tianyu Cui
cuitianyu@iie.ac.cn