Internet Traffic Classification for Faster and Secured Network Service

Purna Chandra Sethi
PhD Scholar, Dept. of Computer Science
Utkal University
Bhubaneswar, India

Prafulla Kumar Behera
Reader, Dept. of Computer Science
Utkal University
Bhubaneswar, India

ABSTRACT
Due to the growth in prominence of Web, there is a need for proficient system administration. Network visibility becomes very crucial for traffic engineering and network management. A large number of users demands varied information at a given time. By identifying the users that demand same type of information and clustering them into different groups, the Internet accessibility and resource utilization can be improved. The most popular solutions for network management are Deep Packet Inspection algorithm, In-Depth Packet Inspection algorithm and some related statistical classification technologies. All these solutions depend on the availability of a training set. Supervised (classification) and unsupervised (clustering) algorithms are used for identification of the network traffic. Network traffic analysis always depends on various parameters such as the data to be searched, the time of searching, available bandwidth, number of accessing users, architecture of the network system, etc. For simplicity, the type of data and the data rate was considered for this implementation. Due to clustering, automatic identification of the classes of traffic was achieved. Since clustering technique is used for group processing of information, group signature techniques is being applied here for secured data processing.

Keywords
Traffic classification, SeLeCT using self-seeding, GFGS Algorithm, SHA-256.

1. INTRODUCTION
In present scenario, almost all operations are web based. Various protocols are being used for the management of varying degree of web traffic. During traffic analysis, the protocols along with its related information are to be identified. This identification phase is the most difficult task of network management and traffic analysis. With proper analysis of traffic, the network can be managed efficiently and consequently can help to enhance the processing speed. Two approaches namely clustering (supervised) and classification (unsupervised) techniques are used for this purpose. Further, for classification of information, the most commonly used algorithm are port based, deep packet inspection and in depth packet inspection. These approaches are pertinent for a fixed trained data set. Any information which remains out of consideration during classification phase are mislabeled which leads to incorrect traffic analysis.

For the above reason, clustering concept is used. Various clustering algorithms are present which differs by their complexity. A new unsupervised algorithm called SeLeCT was proposed [1] for implementation over a large data set consisting of the traces collected in different years from various ISPs located in 3 different continents. SeLeCT gives good level of system deceivability for disclosure of new classes of activity for system administration. The flood of information stream undergoes grouping after a versatile or dynamic seeding methodology for movement examination. The portion size and the inter-arrival time were considered for the investigation. The clusters generated using SeLeCT are nearly 100% pure and homogenous.

In paper [2], the authors proposed a novel Enhanced Hierarchical Multi-pattern Matching Algorithm (EHMA) for packet inspection of application-layer information. Instead of performing exact string matching between packets and a large set of predefined patterns, the paper was based on the frequency of common grams for storage as well as retrieval of information. Common grams are the elements which occur more frequently in the pattern. EHMA was proposed using two-tier architecture. The cluster-wise pattern matching algorithm significantly reduced the amount of external memory accesses as well as on chip memory space. The algorithm was proposed using skip strategy, so that each element of the list does not need to be compared during the searching process. Hence performance of the system increases significantly in terms of reduction in external memory space, on chip memory space and searching time.

In paper [3], the authors proposed a traditional approach for data transmission and guarantees that there will be 100% error free data transmission. When any frame is corrupted, the appropriate frame is selected and retransmitted again. If the frequency of such corrupted information increases, the processing time will increase significantly. Hence there is significant decrease in the efficiency of system. So the proposed algorithm of this paper was applied for security of information so that there would be less chance of information being corrupted.

In paper [4], the author proposed an efficient algorithm for optimization of resource utilization applied in paper [3]. UPnP protocol is used for universal plug and play operation. Since each device and resource is not used at each time, UPnP protocol is applied that allows the devices to be plugged into the network first for use. Then it is unplugged from the network so that the resource utilization will be optimum.

In paper [5], the authors applied an efficient algorithm which can reduce the on-chip memory space efficiently with faster searching approach. Paper [6] is about SHA–256 algorithm which was later known as Advanced Encryption Standard (AES). AES is one of the efficient algorithms that are very difficult to be cracked. It needs sixteen rounds for encryption as well as decryption of information. In paper [7], the authors proposed an efficient algorithm by combining the feature of paper [5] and [6] so that information security will be maintained along with the reduction of on chip memory space. The proposed three tire model enhances the information security along with the faster

searching process. The SHA-256 algorithm implementation makes the information more secure. The data scrambled utilizing SHA-256 is difficult to break. It is essentially a 256-bit block cipher algorithm which encrypts the intermediate hash value using the message block as key.

SHA-256 was applied in infinite field. So, by defining the problem in terms of modulo arithmetic, a set of finite classes called Galois field was generated which works with same complexity along with the improvement in information security as compared to the traditional algorithm.

The rest of this paper is organized as follows: Section 2 presents the related works, previous proposed pattern matching algorithms and the fundamental definitions. Section 3 contains the proposed model applied for Internet traffic analysis using self-seeding approach. SHA–256 cryptography algorithm implementation using EHMA will enhance the security of the proposed algorithm. Section 4 contains the proposed algorithm, Section 5 contains the experimental result, Section 6 presents the performance of algorithm, Section 7 provides the conclusion, and Section 8 provides the future work for this research work.

2. LITERATURE OVERVIEW

2.1 Traffic classification

Network traffic or **data traffic** occurs due to the available data in a network. In computer networks, the data is encapsulated in network packets. Network traffic control, managing, prioritizing, controlling or reducing the network traffic is the major aim of the traffic management. Network traffic measurement and sorting of traffic on a specific system, are the significant issues among network traffic investigation. Network traffic simulation is done to quantify the productivity of a network system. The network traffic relies on different traffic characteristics. The *traffic attributes* provide a mechanism for associating data with a subscriber or with a subscriber's sessions. The attributes that are attached to the subscribers are used to trigger the traffic management policy. Some sample attributes used during traffic analysis:

- Service tier
- Video codec
- IP addresses
- Audio codec
- MAC addresses
- Client device
- Client device type
- Content (e.g., video, audio) provider
- Operating system
- Browser
- Media stream type
- Session protocol
- Media container
- Transport protocol
- Video resolution, etc.

Accurate traffic identification and insightful measurements forms the foundation of network business intelligence and network policy control. Without identifying and measuring parameters of the traffic flow in the network, it is not possible to optimize the network traffic. The tolerance levels are identified for accuracy of the data transmission. Now a day, the traffic classification goes beyond identification (i.e., determining what the traffic is) and extends into extracting information (e.g., video resolution, media type, etc.) and measuring characteristics (e.g., duration, QoS, etc.). Many techniques exist to identify traffic and extract additional information or measure quantities, ranging from relatively simple (e.g., regular expressions) to extremely complex (e.g., stateful trackers and analyzers) parameters. In general, advanced techniques provide the most comprehensive information which is processor-intensive. So, they use DPI policy for network traffic control.

For an extensive traffic classification, the trade-off between the different attributes of packets is done. The traffic classification must focus on the accuracy first along with level of traffic tolerance.

2.2 SeLeCT and self-seeding

SeLeCT, is a Self-Learning Classifier. It is one of the productive procedures utilized for Internet Traffic investigation. It uses unsupervised algorithm with an *adoptive seeding approach* to automatically identify the class of traffic. SeLeCT neither follows the earlier learning of the data nor it make grouping for the system movement. It performs a measurable examination for utilizing for the activity in an automated fashion so that firmly 98% exactness is achieved during networking.

The adoptive seeding technique manages the dynamic data set, so that the clustering will be done in a mechanized manner. The clusters will automatically adopt the new information when it is encountered during processing. Due to the adoptive seeding technique, the elements will be automatically switched from one window to the next. The movement of data and the enlargement of cluster size were represented as given in the figure–1.

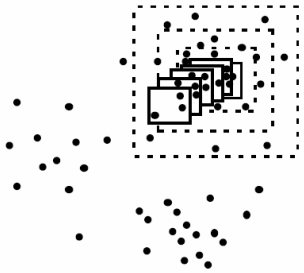

Fig.1 [Movement of data and Enlargement of window]

Some of the key features of SeLeCT are:

Adaptive classification mode: A semi-supervised learning approach allows SeLeCT to learn information from random data set providing automatic traffic management.

Simple iterative approach: SeLeCT is based on k-mean clustering technique. It utilizes k-mean algorithm as a building block in an iterative bunching refinement process, which permits utilizing particular Internet movement components, for example, the server port that can't be coordinated into established grouping calculations in a direct manner. This methodology yields firmly durable groups and gives a practically finish scope of the considered streams.

Leverages layer-4 features: SeLeCT depends on the accessibility of stream level components that effortlessly gained toward the start of the stream, and it doesn't accept to see both directions of traffic.

Limited complexity: SeLeCT can keep running progressively by continually checking the approaching activity, making clusters of streams, and preparing these groups before the next batch accumulates.

2.3 GFGS

In general, the individual packets are considered for data transmission. Individual data transmission will provide overhead of system. So, instead of sending the individual packet, the frequent common gram is chosen which is utilized for sending the individual data. The frequent common grams selected are considered as the representatives of the clustered elements. In the high-layer intrusion detection, patterns may appear anywhere in the packet payload, making the attacking packets difficult to recognize. GFGS expect that a little arrangement of marks be found from the examples, so that the suspicious substrings of T would be simpler to recognize from the innocent parts, and the pattern matching becomes faster. A set of significant grams is defined as representatives of a pattern set P. The 2-gram methodology is applied for discovering the generalized frequent common gram. Any suspicious 2-gram set is selected for each pattern. The frequency of the 2-grams is calculated. The elements with highest frequency maintained in the on-chip cache which is compared for accessing all type of data for storage as well as receive operation [7]. The paper is implemented using the Wholesalers data set of UCI repository. It produces the frequent 2-grams for storage of similar type of information. This is represented as given in the fig-2.

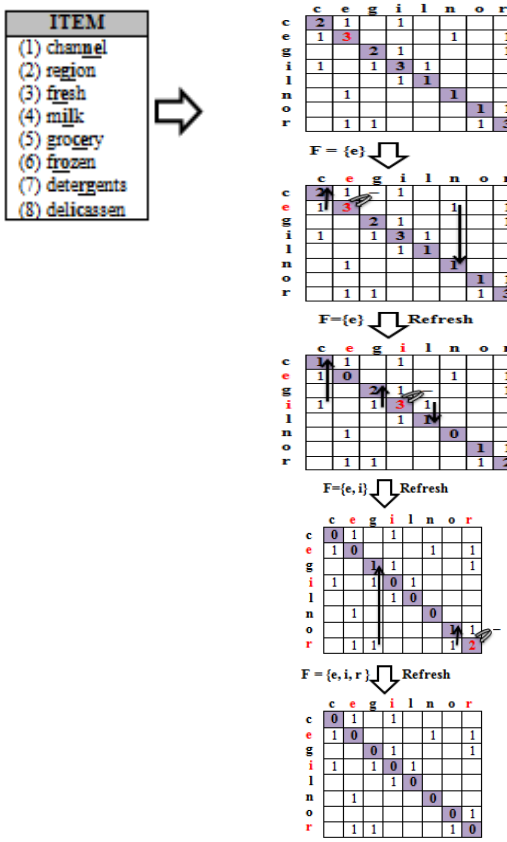

Fig – 2: Example of GFGS

2.4 SHA-256

It is one of the most secure algorithms for data encryption. It is considered as the Advanced Encryption Standard (AES) and was applied for large number of real time applications. This algorithm is nearly impossible to decrypt. Computation of hash for a message is done the following model given below:

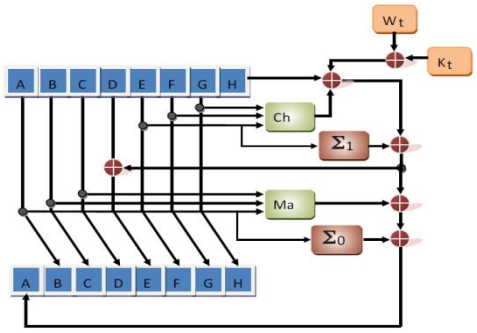

Fig – 3 [Model for SHA – 256]

A sequence of constant words, K_0, K_1, … …, K_{63} is used in SHA-256. These are generated by the first thirty-two bits of the fractional parts of the cube roots of the first sixty-four primes.

3. PROPOSED WORK

The proposed working model for efficient Internet traffic classification stands on four basic pillars such as Traffic classification, GFGS algorithm, SeLeCT algorithm and SHA-256 (Secured Hash Algorithm). The Internet traffic classification is done following various parameters. For simplicity, the type of data and the data flow rate is considered as the parameters. GFGS algorithm produces "frequent common grams" for maintaining cluster elements. SeLeCT algorithm is then applied, following the self-seeding approach for redistribution of cluster elements. Finally, SHA–256 algorithm is applied for the security of data involved. The proposed model is based on the following flow chart.

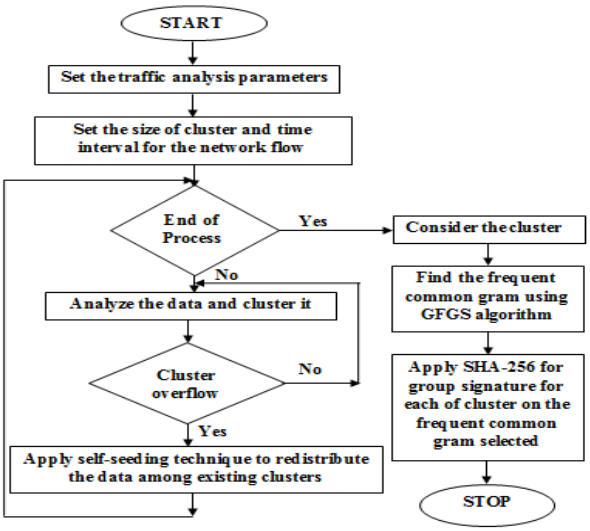

Fig – 4 [Flow chart of the Proposed Model]

3.1 Implementation and Experimental Result

The examination is done following the Clickstream information idea. Clickstream information was produced by

the considering the numbers of clicks done on different web pages by various users in World Wide Web (WWW). Clickstream data is used by Internet marketers and advertisers. Higher click denotes the higher demand for the web page. So, the web pages with higher click rate have to be managed for efficient processing. An example of genuine clickstream records is the MSNBC dataset, portrays the page visits of clients who went by msnbc.com on a solitary day. There are 989,818 clients and just 17 distinct items, in light of the fact that these things are recorded at the level of URL class, not at page level, which enormously lessens the dimensionality. The 17 classes are arranged with their classification number. The data set is represented as:

Front Page	1
News	2
Tech	3
Local	4
Opinion	5
On-air	6
Misc	7
Weather	8
Health	9
Living	10
Business	11
Sports	12
Summary	13
Bbs	14
Travel	15
msn-news	16
msn-sports	17

Table 1: MSNBC Dataset

The sample sequences for the data set will be:

1 1

2

3 2 2 4 2 2 2 3 3

5

1

6

6 7 7 7 6 6 8 8 8 8

6 9 4 4 4 10 3 10 5 10 4 4 4

1 1 1 1 1 1 1 1

Each row describes the hits of a single user. For example, the first user hits "frontpage" twice, and the second user hits "news" once. Similarly third user hits “Tech” page once, “News” twice, then “Local” page once and then again “News” page twice. Finally, it accesses “Tech” page twice. The frequency of accessibility is represented as:

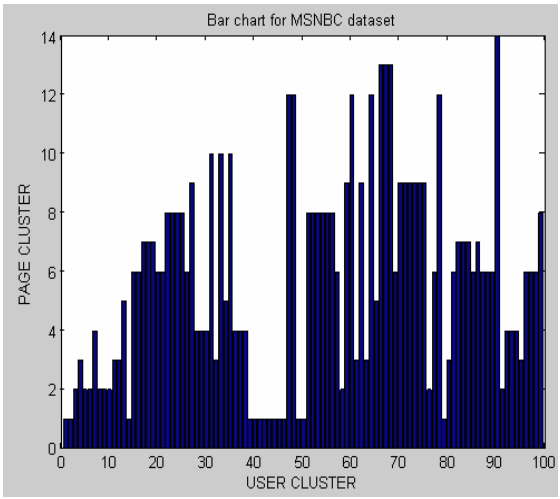

Fig – 5 [MSNBC Data set representation]

3.2 Experimental Setup

The algorithm is implemented using C++ in Intel core 2 duo 2.20GHz speed processor, 4 GB RAM. The MSNBC data set is used for implementation of the algorithm.

4. PROPOSED ALGORITHM

The proposed algorithm for secured network traffic analysis based on clustering approach is defined as:

Step 1: Set the traffic Analysis Parameter

Step 2: Set the maximum size of the cluster

Step 3: Set the time interval for the network flow

Step 4: Send the block of information sequentially at the specified time interval.

Step 5: while (!EOF) do

- 5.1. Analyze the data and cluster it
- 5.2. IF (Cluster overflow condition) Then
- 5.2.1. Apply self-seeding technique to redistribute the data among the existing clusters
- 5.2.2. Goto Step 5
- 5.3. ELSE
- 5.3.1. Goto Step 5.1.

Step 6:

- 6.1. Consider the set of clusters
- 6.2. Find the Generalized frequent common gram
- 6.3. Store the information having similar property using the frequent common grams.
- 6.4. Apply SHA-256 algorithm on the generalized frequent common gram to generate group signature for individual cluster

5. EXPERIMENTAL RESULT

During experiment, let the number of clicks are considered as the major parameter for clustering. For the first click, the web page will be loaded into the first cluster. When the numbers

clicks increases then the web pages will move to any other cluster. Total four clusters are considered for the experiment of the proposed model.

The set of 10 units of data are considered for the clustering. The experimental result after clustering will be represented as given in table–2.

Based on the result of the table–2, the SHA-256 algorithm is applied on the frequent common grams. Due to such implementation, group signature is created by applying the SHA–256 algorithm on the frequent common grams, so that the different cluster elements can't be accessed directly. The hash keys produced are considered as the group signature for the corresponding cluster. The SHA–256 hash key for each element of the data set is given in table–4.

Due to simple clustering implementation, the first cluster overhead occurs, i.e. maximum of data are stored in the first cluster. So, an automated redistribution technique is applied to reduce the overhead of first cluster and transfer them to other clusters. Considering the maximum size of cluster as 25, the redistribution scheme is represented by the table–3.

6. PERFORMANCE OF ALGORITHM

- GFGS algorithm provides the generalized frequent common gram. The frequent common behaves as the representatives for each cluster. By maintaining the frequent common grams only, maintenance of cluster is done easily.
- The GFGS algorithm implementation reduces the searching of whole information. By selecting the frequent common grams, the clusters elements are directly accessed which reduces the searching time.
- By self-seeding approach, when a cluster overflow occurs instead of creation of new cluster, a redistribution scheme is applied. This method reduces the overhead of clusters.
- SHA-256 uses a cryptographic hash functions that runs on digital data and by comparing the result of the "hash" to a known and expected hash value, a person can determine the data authenticity. After authentication verification, the information is accessed. SHA-256 implementation on the frequent common grams will provide more security to the information.
- The added benefit of cryptographic hash functions using SHA–256 is they are almost impossible to reverse engineer to reconstruct the original data. Hence the method is more secure.

7. CONCLUSION

Sequential searching of data in random data set requires more searching time. By the application of clustering techniques, the data having similar properties are kept in distinct clusters so that searching becomes faster and maintenance becomes easier. In clustering algorithms, since each element of clusters is considered separately, it increases the complexity of the process. To make the process easier, self-seeding approach is being used to redistribute data among the clusters. The data is redirected to different cluster having similar features so that load balancing among the clusters is maintained. With the help of GFGS algorithm, the frequent common grams are selected for each of the cluster. The cluster elements are identified using selected frequent common grams. This procedure helps to reduce searching time significantly as compared to sequential search. Finally, SHA-256 algorithm is being applied on the frequent common grams to generate the hash key used for storing the clustered information. The resultant data is compared with the result of the "hash" to a pre-defined hash value for verifying the data authenticity. The information is being allowed to be accessed only after its authentic verification. The added benefit of cryptographic hash functions using SHA–256 is that they are very difficult to reverse engineer for reconstructing the original data. Hence this method is more secure as compared to earlier methods.

Most of the cryptography algorithms involve complex mathematical operations, which needs exponential time to crack using a general purpose computer. As a future scope, if all the cryptography processes can be defined in terms of light (electric signal) then processing speed will be faster. This may be achieved by the digital implementation in terms of light which may lead to faster processing for encryption as well as decryption.

8. REFERENCES

[1] Luigi Grimaudo, Marco Mellia, Elena Baralis and Ram Keralapura, "SeLeCT: Self-Learning Classifier for Internet Traffic", IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, VOL. 11, NO. 2, JUNE 2014 (P144 – P157)

[2] Tzu-Fang Sheu, Nen-Fu Huang, and Hsiao-Ping Lee, "In-Depth Packet Inspection Using a Hierarchical Pattern Matching Algorithm", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 7, NO. 2, APRIL-JUNE 2010

[3] P. C. Sethi, C. Dash: "High Impact Event Processing using Incremental Clustering in Unsupervised Feature Space through Genetic algorithm by Selective Repeat ARQ protocol", ICCCT- 2nd IEEE Conference – 2011, pp. 310-315.

[4] P. C. Sethi, "UPnP and Secure Group communication Technique for Zero-configuration Environment construction using Incremental Clustering", International Journal of Engineering Research & Technology (IJERT), Vol. 2 Issue 12, December – 2013, ISSN: 2278-0181, pp. 2095-2101

[5] "SHA-256" wikipedia.org SHA-2.

[6] P. C. Sethi, P.K. Behera, "Secure Packet Inspection using Hierarchical Pattern matching implemented Using Incremental Clustering Algorithm", December-22-24, ICHPCA-2014 (IEEE International Conference)

9. AUTHOR PROFILE

Purna Chandra Sethi received the B. Tech and M.Tech degrees in Information Technology Engineering and Computer Science Engineering from College of Engineering & Technology, Bhubaneswar. He has qualified UGC-NET three times in Computer Science and Applications. He is currently pursuing PhD in P.G. Department of Computer Science at Utkal University, Odisha, India. His current research area of interest is Network Security and QoS. He is a life time member of CSI, ISTE, IAENG, CSTA.

Dr. P. K. Behera is currently working as Reader at Department of Computer Science, Utkal University, Bhubaneswar, Odisha, India. He has more than two decades of teaching experience. His area of interest is MANET, Wireless Network, Distributed Systems, Mobile Computing,

Network and Information Security, Software Engineering. He has published number of research papers in reputed International Conferences and Journals. He is a reviewer of many national and International referred Journals. He is the Secretary of CSI Bhubaneswar Chapter.

10. APPENDIX

Table–2 [Clustering without self-seeding method]

Size of data set	No. of elements in cluster1	Frequent common gram for cluster1	No. of elements in cluster2	Frequent common gram for cluster2	No. of elements in cluster3	Frequent common gram for cluster3	No. of elements in cluster4	Frequent common gram for cluster4
10	10	2	0	-	0	-	0	-
20	13	2	7	6	0	-	0	-
30	16	2	13	6	1	9	0	-
40	22	4	14	6	4	10	0	-
50	30	1	14	6	6	10	0	-
60	31	1	21	8	8	10	0	-
70	33	1	23	8	11	12	3	13
80	36	1	24	8	17	9	3	13
90	36	1	33	6	17	9	4	13
100	42	1	37	6	17	9	4	13

Table–3 [Clustering using self-seeding method]

Size of data set	No. of elements in cluster1	Frequent common gram for cluster1	No. of elements in cluster2	Frequent common gram for cluster2	No. of elements in cluster3	Frequent common gram for cluster3	No. of elements in cluster4	Frequent common gram for cluster4
10	10	2	0	-	0	-	0	-
20	13	2	7	6	0	-	0	-
30	16	2	13	6	1	9	0	-
40	22	4	14	6	4	10	0	-
50	25	4	19	6	6	10	0	-
60	25	1	25	8	10	10	0	-
70	25	1	25	8	17	12	3	13
80	25	1	25	8	25	9	5	13
90	25	1	25	8	25	9	15	6
100	25	1	25	8	25	9	25	6

Table–4 [Hash key for the different types of Frequent common grams]

Data Value	Hash key value after SHA–256 Implementation
1	6b86b273ff34fce19d6b804eff5a3f5747ada4eaa22f1d49c01e52ddb7875b4b
2	d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f90da3a666eec13ab35
3	4e07408562bedb8b60ce05c1decfe3ad16b72230967de01f640b7e4729b49fce
4	4b227777d4dd1fc61c6f884f48641d02b4d121d3fd328cb08b5531fcacdabf8a
5	ef2d127de37b942baad06145e54b0c619a1f22327b2ebbcfbec78f5564afe39d
6	e7f6c011776e8db7cd330b54174fd76f7d0216b612387a5ffcfb81e6f0919683
7	7902699be42c8a8e46fbbb4501726517e86b22c56a189f7625a6da49081b2451
8	2c624232cdd221771294dfbb310aca000a0df6ac8b66b696d90ef06fdefb64a3
9	19581e27de7ced00ff1ce50b2047e7a567c76b1cbaebabe5ef03f7c3017bb5b7
10	4a44dc15364204a80fe80e9039455cc1608281820fe2b24f1e5233ade6af1dd5
11	4fc82b26aecb47d2868c4efbe3581732a3e7cbcc6c2efb32062c08170a05eeb8
12	6b51d431df5d7f141cbececcf79edf3dd861c3b4069f0b11661a3eefacbba918
13	3fdba35f04dc8c462986c992bcf875546257113072a909c162f7e470e581e278
14	8527a891e224136950ff32ca212b45bc93f69fbb801c3b1ebedac52775f99e61
15	e629fa6598d732768f7c726b4b621285f9c3b85303900aa912017db7617d8bdb
16	b17ef6d19c7a5b1ee83b907c595526dcb1eb06db8227d650d5dda0a9f4ce8cd9
17	4523540f1504cd17100c4835e85b7eefd49911580f8efff0599a8f283be6b9e3