

CLAIMS

What is claimed is:

1. A method comprising:

performing at a client computer system an initial malware scan of a plurality of target objects of the client computer system; and

in response to a tentative determination by the initial malware scan that the target object is suspected of being malicious:

generating at the client computer system a plurality of target hashes of the target object, each target hash representing a distinct code block of the target object, each distinct code block consisting of a sequence of processor instructions of the target object;

sending from the client computer system the plurality of target hashes to a server computer system connected to the client computer system over a wide area network; and

receiving at the client computer system from the server computer system a server-side indicator of whether the target object is malicious, wherein the server-side indicator is generated by the server computer system by:

for at least a target hash of the plurality of target hashes, retrieving a plurality of reference hashes of a reference object, the reference object selected from a set of whitelisted objects according to the target hash, and when the plurality of target hashes is not identical to the plurality of reference hashes, determining a similarity score according to a count of hashes common to both the plurality of target hashes and the plurality of reference hashes; and

when the similarity score exceeds a predetermined threshold, designating the target object as non-malicious.

- 1 2. The method of claim 1, wherein generating the server-side indicator by the computer
2 server system comprises:
3 when the similarity score does not exceed the predetermined threshold, generating a
4 filtered set of target hashes of the target object by filtering all target hashes
5 which appear in a database of clean hashes out of the plurality of target hashes
6 of the target object; and
7 comparing the filtered set of target hashes to a database of malware-identifying
8 hashes specific to malware.
- 1 3. The method of claim 1, wherein generating the server-side indicator by the computer
2 server system comprises:
3 when the similarity score does not exceed the predetermined threshold, generating a
4 filtered set of target hashes of the target object by filtering all target hashes
5 which appear in a database of clean hashes out of the plurality of target hashes
6 of the target object; and
7 comparing the filtered set of target hashes to a database of outbreak-detection hashes
8 specific to unknown objects reported within a predetermined recent period by
9 a plurality of distinct client computer systems connected to the server
10 computer system.
- 1 4. A method comprising:
2 receiving at a server computer system a plurality of target hashes of a target object of a
3 client computer system connected to the server computer system over a wide area
4 network, wherein the plurality of target hashes are generated at the client
5 computer system in response to a tentative determination by the client computer
6 system that the target object is suspected of being malicious, the tentative
7 determination resulting from an initial malware scan of a plurality of target
8 objects of the client computer system;
9 generating at the server computer system a server-side indicator of whether the target
10 object is malicious by:

11 for at least a target hash of the plurality of target hashes, retrieving a plurality of
12 reference hashes of a reference object, the reference object selected from a
13 set of whitelisted objects according to the target hash, and when the
14 plurality of target hashes is not identical to the plurality of reference
15 hashes, determining a similarity score according to a count of hashes
16 common to both the plurality of target hashes and the plurality of
17 reference hashes, and
18 when the similarity score exceeds a predetermined threshold, designating the
19 target object as non-malicious; and
20 sending to the client computer system the server-side indicator of whether the target
21 object is malicious.
22

- 1 5. The method of claim 4, wherein generating the server-side indicator by the computer
2 server system comprises:
3 when the similarity score does not exceed the predetermined threshold, generating a
4 filtered set of target hashes of the target object by filtering all target hashes
5 which appear in a database of clean hashes out of the plurality of target hashes
6 of the target object; and
7 comparing the filtered set of target hashes to a database of malware-identifying
8 hashes specific to malware.
9
- 1 6. The method of claim 4, wherein generating the server-side indicator by the computer
2 server system comprises:
3 when the similarity score does not exceed the predetermined threshold, generating a
4 filtered set of target hashes of the target object by filtering all target hashes
5 which appear in a database of clean hashes out of the plurality of target hashes
6 of the target object; and
7 comparing the filtered set of target hashes to a database of outbreak-detection hashes
8 specific to unknown objects reported within a predetermined recent period by
9 a plurality of distinct client computer systems connected to the server
10 computer system.

11
1 7. A method comprising:

2 receiving at a server computer system a plurality of target hashes of a target object, each
3 target hash representing a distinct code block of the target object, each distinct
4 code block consisting of a sequence of processor instructions of the target object;
5 for at least a target hash of the plurality of target hashes, employing the server computer
6 system to:

7 retrieve a plurality of reference hashes of a reference object, the reference object
8 selected from a set of whitelisted objects according to the target hash, and
9 when the plurality of target hashes is not identical to the plurality of reference
10 hashes, determine a similarity score according to a count of hashes
11 common to both the plurality of target hashes and the plurality of
12 reference hashes; and

13 when the similarity score exceeds a predetermined threshold, employing the server
14 computer system to label the target object as non-malicious.
15

1 8. The method of claim 7, wherein the target hash comprises a hash of an opcode
2 pattern, the opcode pattern comprising a sequence of instruction indicators, each
3 instruction indicator indicative of a processor instruction of the distinct code block.
4

1 9. The method of claim 7, wherein the sequence of processor instructions consists of
2 between 5 and 50 consecutive processor instructions.
3

1 10. The method of claim 9, wherein the sequence of processor instructions consists of
2 between 5 and 15 consecutive processor instructions.
3

1 11. The method of claim 7, wherein the sequence of processor instructions starts with a
2 CALL instruction.
3
4
5

1 12. The method of claim 7, further comprising:
2 performing a code normalization procedure on the target object to produce a
3 normalized target object, and wherein each distinct code block consists of a
4 sequence of computer instructions of the normalized target object; and
5 applying a hash function to the distinct code block to produce the target hash.
6

1 13. The method of claim 7, wherein the similarity score is determined as a function of:

$$C/\max(N_T, N_R)$$

3 wherein C denotes the count of hashes common to both the plurality of target hashes
4 and the plurality of reference hashes, while N_T and N_R denote the cardinality of the
5 plurality of target hashes and the cardinality of the plurality of reference hashes,
6 respectively.
7

1 14. The method of claim 7, wherein the similarity score is determined as a function of:

$$C/(N_T + N_R)$$

3 wherein C denotes the count of hashes common to both the plurality of target hashes and
4 the plurality of reference hashes, while N_T and N_R denote the cardinality of the plurality
5 of target hashes and the cardinality of the plurality of reference hashes, respectively.
6

1 15. The method of claim 7, wherein the similarity score is determined as a function of:

$$C/N_T + C/N_R$$

3 wherein C denotes the count of hashes common to both the plurality of target hashes and
4 the plurality of reference hashes, while N_T and N_R denote the cardinality of the plurality
5 of target hashes and the cardinality of the plurality of reference hashes, respectively.
6

1 16. The method of claim 7, wherein the target object comprises a computer file.
2

1 17. The method of claim 7, wherein the target object comprises a computer process.
2
3
4

- 1 18. A computer system comprising at least a processor programmed to:
2 receive a plurality of target hashes, each target hash representing a distinct code block of
3 a target object, each distinct code block consisting of a sequence of processor
4 instructions of the target object;
5 for at least a target hash of the plurality of target hashes:
6 retrieve a plurality of reference hashes of a reference object, the reference object
7 selected from a set of whitelisted objects according to the target hash, and
8 when the plurality of target hashes is not identical to the plurality of reference
9 hashes, determine a similarity score according to a count of hashes
10 common to both the plurality of target hashes and the plurality of
11 reference hashes; and
12 when the similarity score exceeds a predetermined threshold, label the target object as
13 non-malicious.
14
- 1 19. The system of claim 18, wherein the target hash comprises a hash of an opcode
2 pattern, the opcode pattern comprising a sequence of instruction indicators, each
3 instruction indicator indicative of a processor instruction of the distinct code block.
4
- 1 20. The system of claim 18, wherein the sequence of processor instructions consists of
2 between 5 and 50 consecutive processor instructions.
3
- 1 21. The system of claim 20, wherein the sequence of processor instructions consists of
2 between 5 and 15 consecutive processor instructions.
3
- 1 22. The system of claim 18, wherein the sequence of processor instructions starts with a
2 CALL instruction.
3
- 1 23. The system of claim 18, wherein the processor is further programmed to:
2 perform a code normalization procedure on the target object to produce a normalized
3 target object, wherein each distinct code block consists of a sequence of
4 computer instructions of the normalized target object; and

5 apply a hash function to the distinct code block to produce the target hash.

6
1 24. The system of claim 18, wherein the similarity score is determined as a function of:

$$C/\max(N_T, N_R)$$

2
3 wherein C denotes the count of hashes common to both the plurality of target hashes
4 and the plurality of reference hashes, while N_T and N_R denote the cardinality of the
5 plurality of target hashes and the cardinality of the plurality of reference hashes,
6 respectively.

7
1 25. The system of claim 18, wherein the similarity score is determined as a function of:

$$C/(N_T + N_R)$$

2
3 wherein C denotes the count of hashes common to both the plurality of target hashes
4 and the plurality of reference hashes, while N_T and N_R denote the cardinality of the
5 plurality of target hashes and the cardinality of the plurality of reference hashes,
6 respectively.

7
1 26. The system of claim 18, wherein the similarity score is determined as a function of:

$$C/N_T + C/N_R$$

2
3 wherein C denotes the count of hashes common to both the plurality of target hashes
4 and the plurality of reference hashes, while N_T and N_R denote the cardinality of the
5 plurality of target hashes and the cardinality of the plurality of reference hashes,
6 respectively.

7
1 27. The system of claim 18, wherein the target object comprises a computer file.

2
1 28. The system of claim 18, wherein the target object comprises a computer process.
2
3
4

1 29. A computer system comprising:

2 means for receiving a plurality of target hashes, each target hash representing a distinct
3 code block of a target object, each distinct code block consisting of a sequence of
4 processor instructions of the target object;

5 means for retrieving a plurality of reference hashes of a reference object, the reference
6 object selected from a set of whitelisted objects according to a selected target hash
7 of the plurality of target hashes;

8 means for determining a similarity score according to a count of hashes common to both
9 the plurality of target hashes and the plurality of reference hashes; and

10 means for labeling the target object as non-malicious according to the similarity score.
11

1 30. A non-transitory computer-readable storage medium encoding instructions which,
2 when executed on a processor, cause the processor to perform the steps of:

3 receiving a plurality of target hashes, each target hash representing a distinct code
4 block of a target object, each distinct code block consisting of a sequence of
5 processor instructions of the target object;

6 for at least a target hash of the plurality of target hashes:

7 retrieving a plurality of reference hashes of a reference object, the
8 reference object selected from a set of whitelisted objects according to the
9 target hash, and

10 when the plurality of target hashes is not identical to the plurality of
11 reference hashes, determining a similarity score according to a count of
12 hashes common to both the plurality of target hashes and the plurality of
13 reference hashes; and

14 when the similarity score exceeds a predetermined threshold, labeling the target object as
15 non-malicious.
16

1 31. A method comprising:

2 receiving at a server computer system a plurality of target hashes, each target hash
3 representing a distinct data block of a target object of a client computer system

4 connected to the server computer system, each distinct code block consisting of a
5 sequence of processor instructions of the target object;

6 in response to receiving the plurality of target hashes, employing the server computer
7 system to retrieve a plurality of reference hashes representing a whitelisted data
8 object, and

9 in response to determining that the plurality of target hashes is not identical to the
10 plurality of reference hashes, and determining that the plurality of target hashes
11 and the plurality of reference hashes share a majority of items, labeling the target
12 object as non-malicious.
13

CLAIMS

1. A method comprising employing at least one processor to perform each of the steps of:
receiving a plurality of target hashes computed for a target data object, each target hash
representing a distinct sequence of processor instructions of the target data object;

5 retrieving a plurality of reference hashes representing a whitelisted data object, and

in response to determining that the plurality of target hashes is not identical to the
plurality of reference hashes, and in response to determining that the plurality of target hashes
and the plurality of reference hashes share a majority of items, labeling the target data object
as non-malicious, wherein labeling the target data object as non-malicious is performed in
10 response to determining a similarity score according to a count of hashes common to both the
plurality of target hashes and the plurality of reference hashes.

2. The method of claim 1, wherein each target hash comprises a hash of an opcode pattern,
the opcode pattern comprising a sequence of instruction indicators, each instruction indicator
indicative of a processor instruction of the distinct sequence of processor instructions.

15 3. The method of claim 1, wherein the distinct sequence of processor instructions consists
of between 5 and 50 consecutive processor instructions.

4. The method of claim 1, wherein the distinct sequence of processor instructions consists
of between 5 and 15 consecutive processor instructions.

20 5. The method of claim 1, wherein the distinct sequence of processor instructions starts
with a CALL instruction.

6. The method of claim 1, further comprising:

performing a code normalization procedure on the target data object to produce a
normalized object, and wherein the distinct sequence of processor instructions consists of a
sequence of processor instructions of the normalized object; and

25 applying a hash function to the distinct sequence of processor instructions to produce
the target hash.

7. The method of claim 1, wherein the target data object comprises a computer file.

8. The method of claim 1, wherein the target data object comprises a computer process.

9. The method of claim 1, wherein retrieving the plurality of reference hashes is performed in response to receiving the plurality of target hashes.

10. A non-transitory computer-readable medium encoding instructions which, when
5 executed by a computer system having at least one processor, cause the computer system to:
receive a plurality of target hashes computed for a target data object, each target hash
representing a distinct sequence of processor instructions of the target data object;

retrieve a plurality of reference hashes representing a whitelisted data object, and

in response to determining that the plurality of target hashes is not identical to the
10 plurality of reference hashes, and in response to determining that the plurality of target hashes
and the plurality of reference hashes share a majority of items, label the target data object as
non-malicious, wherein labeling the target data object as non-malicious is performed in
response to determining a similarity score according to a count of hashes common to both the
plurality of target hashes and the plurality of reference hashes.

11. The computer-readable medium of claim 10, wherein each target hash comprises a hash
15 of an opcode pattern, the opcode pattern comprising a sequence of instruction indicators, each
instruction indicator indicative of a processor instruction of the distinct sequence of processor
instructions.

12. The computer-readable medium of claim 10, wherein the distinct sequence of processor
20 instructions consists of between 5 and 50 consecutive processor instructions.

13. The computer-readable medium of claim 12, wherein the distinct sequence of processor
instructions consists of between 5 and 15 consecutive processor instructions.

14. The computer-readable medium of claim 10, wherein the distinct sequence of processor
instructions starts with a CALL instruction.

15. The computer-readable medium of claim 10, wherein the instructions further cause the
25 computer system to:

perform a code normalization procedure on the target data object to produce a normalized object, and wherein the distinct sequence of processor instructions consists of a sequence of processor instructions of the normalized object; and

5 apply a hash function to the distinct sequence of processor instructions to produce the target hash.

16. The computer-readable medium of claim 10, wherein the target data object comprises a computer file.

17. The computer-readable medium of claim 10, wherein the target data object comprises a computer process.

10 18. The computer-readable medium of claim 10, wherein retrieving the plurality of reference hashes is performed in response to receiving the plurality of target hashes.

19. A computer system comprising at least one processor configured to form:

means for receiving a plurality of target hashes computed for a target data object, each target hash representing a distinct sequence of processor instructions of the target data object;

15 means for retrieving a plurality of reference hashes representing a whitelisted data object, and

means for labeling the target data object as non-malicious in response to determining that the plurality of target hashes is not identical to the plurality of reference hashes and in response to determining that the plurality of target hashes and the plurality of reference hashes share a majority of items, wherein labeling the target data object as non-malicious is performed in response to determining a similarity score according to a count of hashes common to both the plurality of target hashes and the plurality of reference hashes.

20 20. The system of claim 19, wherein retrieving the plurality of reference hashes is performed in response to receiving the plurality of target hashes.