

CLAIMS:

1. A method, comprising:

discovering, by a network monitoring apparatus, nodes in a network;
collecting a list of internet protocol (IP) addresses, media access control (MAC) addresses, domain name system (DNS) names, and sysnames for each of the nodes discovered in the network;

comparing, by an IP duplicate detector, the IP addresses of each of the discovered nodes with IP addresses of current nodes and other discovered nodes;

comparing, by a MAC duplicate detector, the MAC addresses of each of the discovered nodes with MAC addresses of the current nodes and the other discovered nodes;

comparing, by a DNS duplicate detector, the DNS names of each of the discovered nodes with DNS names of the current nodes and the other discovered nodes;

comparing, by a name duplicate detector, the sysnames of each of the discovered nodes with sysnames of the current nodes and the other discovered nodes;

determining duplicate nodes that are duplicates of the other discovered nodes and/or the current nodes based on the comparison of the IP addresses, MAC addresses, DNS names, and sysnames; assigning a priority to each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector that determines an order of execution of each of the comparing steps; and assigning a weight to each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector, wherein the weight indicates a reliability of a result provided by each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector.

2. The method according to claim 1, further comprising discarding the duplicate nodes.

3. The method according to claim 1, wherein the determining comprises executing the following formula:

$$d_1.\text{IsDuplicate()}*d_1.\text{Priority}+\dots+d_n.\text{IsDuplicate()}*d_n.\text{Priority}.$$

4. The method according to claim 1, wherein each of the discovered nodes is assigned a node ID, and wherein the method further comprises assigning a MatchIndex to each node ID, the MatchIndex indicating a likelihood of a match between the discovered node and any of the current nodes and the other discovered nodes.

5. The method according to claim 4, further comprising grouping the duplicate nodes by node ID and summing the MatchIndexes for the same node ID.

6. An apparatus, comprising:

at least one processor and at least one memory including computer program code,

the at least one memory and the computer program code are configured, with the at least one processor, to cause the apparatus at least to discover nodes in a network;

collect a list of internet protocol (IP) addresses, media access control (MAC) addresses, domain name system (DNS) names, and sysnames for each of the nodes discovered in the network;

wherein the at least one processor is further configured to execute:

an IP duplicate detector configured to compare the IP addresses of each of the discovered nodes with IP addresses of current nodes and other discovered nodes;

a MAC duplicate detector configured to compare the MAC addresses of each of the discovered nodes with MAC addresses of the current nodes and the other discovered nodes;

a DNS duplicate detector configured to compare the DNS names of each of the discovered nodes with DNS names of the current nodes and the other discovered nodes;

a name duplicate detector configured to compare the sysnames of each of the discovered nodes with sysnames of the current nodes and the other discovered nodes;

wherein the at least one memory and the computer program code are further configured, with the at least one processor, to cause the apparatus at least to

determine duplicate nodes that are duplicates of the other discovered nodes and/or the current nodes based on the result of comparison of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector; assign a priority to each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector that determines an order of execution; and assign a weight to each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector, wherein the weight indicates a reliability of a result provided by each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector.

7. The apparatus according to claim 6, wherein the at least one memory and the computer program code are further configured, with the at least one processor, to cause the apparatus at least to discard the duplicate nodes.

8. The apparatus according to claim 6, wherein the at least one memory and

the computer program code are further configured, with the at least one processor, to cause the apparatus at least to determine the duplicate nodes by executing the following formula:

$$d_1.IsDuplicate()*d_1.Priority+...+d_n.IsDuplicate()*d_n.Priority.$$

9. The apparatus according to claim 6, wherein each of the discovered nodes is assigned a node ID, and wherein the at least one memory and the computer program code are further configured, with the at least one processor, to cause the apparatus at least to assign a MatchIndex to each node ID, the MatchIndex indicating a likelihood of a match between the discovered node and any of the current nodes and the other discovered nodes.

10. The apparatus according to claim 9, wherein the at least one memory and the computer program code are further configured, with the at least one processor, to cause the apparatus at least to group the duplicate nodes by node ID and to sum the MatchIndexes for the same node ID.

11. The apparatus according to claim 6, wherein a weight is assigned to each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector, wherein the weight indicates the reliability of the result provided by the respective duplicate detectors.

12. A computer program, embodied on a non-transitory computer readable medium, wherein the computer program is configured to control a processor to execute a process, comprising:

- discovering, by a network monitoring apparatus, nodes in a network;
- collecting a list of internet protocol (IP) addresses, media access control (MAC) addresses, domain name system (DNS) names, and sysnames for each of the nodes discovered in the network;

comparing, by an IP duplicate detector, the IP addresses of each of the discovered nodes with IP addresses of current nodes and other discovered nodes;

comparing, by a MAC duplicate detector, the MAC addresses of each of the discovered nodes with MAC addresses of the current nodes and the other discovered nodes;

comparing, by a DNS duplicate detector, the DNS names of each of the discovered nodes with DNS names of the current nodes and the other discovered nodes;

comparing, by a name duplicate detector, the sysnames of each of the discovered nodes with sysnames of the current nodes and the other discovered nodes;

determining duplicate nodes that are duplicates of the other discovered nodes and/or the current nodes based on the comparison of the IP addresses, MAC addresses, DNS names, and sysnames; and assigning a priority to each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector that determines an order of execution of each of the comparing steps; and assigning a weight to each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector, wherein the weight indicates a reliability of a result provided by each of the IP duplicate detector, the MAC duplicate detector, the DNS duplicate detector, and the name duplicate detector.