IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 29, 2020, accepted October 20, 2020, date of publication October 23, 2020, date of current version November 5, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3033391

PTVis: Visual Narrative and Auxiliary Decision to Assist in Comprehending the Penetration Testing Process

SIJIE ZHENG[1], YADONG WU[2], (Member, IEEE), SONG WANG[1], YONG WEI[1], DONGSHENG MU[1], HUAN HE[1], DONGXUAN HAN[1], JING LIAO[1], AND HUARONG CHEN[1]

[1]School of Computer Science and Technology, Southwest University of Science and Technology, Mianyang 621000, China
[2]School of Computer Science and Engineering, Sichuan University of Science and Engineering, Zigong 643002, China

Corresponding author: Yadong Wu (wyd028@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61802320 and Grant 61872304, in part by the National Key Research and Development Project of China under Grant 2016QY04W0801, and in part by the National Defense Pre-Research Foundation of China under Grant JCKY2018404C001 and Grant JCKY2019204B007.

ABSTRACT Due to the complexity of network penetration and the diversity of penetration methods, traditional analysis approaches analyse only a single penetration method or part of the network penetration process. Moreover, the lack of customized exploration makes it difficult to discover and analyse network penetration behaviors. Characterizing and summarizing the penetration testing process based on an interpretive visual analysis approach can enhance researchers' comprehension of penetration testing and further promote the development of network security technologies. To assist with this process, we design PTVis, a visual approach for the penetration testing process summarization based on visual narrative and auxiliary decision. PTVis consists of two primary components: (1) a visual interface that displays customized penetration testing paths, and (2) a component that effectively displays the results of penetration testing. To design PTVis, penetration testing paths that combine penetration testing methods and tools are built via cooperative multi-view and customized exploration, which facilitates the exploration of penetration testing. For evaluation, a qualitative user study is performed on two groups. The feedback from the study demonstrates that PTVis can enhance the user's knowledge of the penetration testing process.

INDEX TERMS Visual analysis, penetration testing, customized exploration, visual narrative, auxiliary decision.

I. INTRODUCTION

Finding the worst cyber threat culprits through security assessment is of great significance for network security [1]. Penetration testing is a branch of security assessment, that originated from military network attacks and defence technology [2]. Penetration testing is a kind of beneficial supplement to network security assessment. It aims to check and audit a computer network system as a simulated attacker [3]. The study of the penetration testing process can strengthen the network defence capabilities of an organization, reduce network security risks, and assist security analysts in comprehending new network attack techniques. With the increasing complexity of computing environments and the diversity of network penetration methods in large organizations, it is very challenging for analysts to perform thorough, accurate, and timely analyses of network penetration activities. Therefore, it is necessary to assist security analysts in summarizing the main characteristics of the penetration testing activities to improve their comprehension of penetration testing.

One way of helping to comprehend something is to summarize its characteristics [4]. An automatic, text-based summary provides a concise method [5] for describing the main content of the penetration testing process. However, this method fails to describe a more in-depth, multifaceted context. For example, "What is the magnitude of the data traffic between penetration testing steps? What are the connections between penetration testing stages?" The drawbacks of the above method limit its overall utility. In contrast, a visual-based penetration testing summary can provide a more effective

The associate editor coordinating the review of this manuscript and approving it for publication was Songwen Pei.

method. Previous work in the field of visual analysis of penetration testing has mostly focused on the analysis of network penetration events (e.g., [6], [7]). With the improvement in network penetration methods, to promote research on new attack technologies and to have a deeper comprehension of network penetration activities, an increasing amount of work is focusing on the visual analysis of penetration testing processes and methods (e.g., [8], [9]). However, there is still a lack of systematic research in terms of visualizing the process of penetration testing and analysing the main characteristics of penetration testing activities. Based on this research motivation, we designed PTVis: a lightweight, intuitive, and informative visual interface for describing and summarizing the main characteristics of penetration testing. By simulating penetration testing scenarios, PTVis helps network security analysts to better comprehend the effectiveness and efficiency of penetration testing activities. PTVis helps professional penetration testers communicate with stakeholders, and it further enhances decision-makers' awareness of network security.

To develop PTVis, first, we discussed the data requirements that reveal the main characteristics of the penetration testing process with experts in the penetration testing field. Second, based on the display requirements of the data, the design decision was made, and the interpretative mapping of the penetration testing was constructed. Then, a visual presentation form of a penetration testing path based on the auxiliary decision and a penetration testing result display interface for retrospective analysis were designed. To evaluate the effectiveness of PTVis in improving the comprehension of users, we conducted user research on two groups of participants. The research results showed that PTVis can help users obtain information and knowledge of penetration testing activities and that it can help users to better understand the effectiveness and necessity of penetration testing through the visual narrative.

Based on the characteristics and analysis points of the penetration testing process, the PTVis analysis process is designed, as shown in Figure 1. After the penetration testing step and tool data were processed, the interpretive elements were extracted to construct penetration testing-interpretive mapping. Using the visual model, users can construct customized penetration testing paths and explore the characteristics and details of the penetration testing process. In addition, users can complete different analysis tasks during the interaction process, thereby enhancing their comprehension of the penetration testing process.

The main contributions of our work include the following:

- **A systematic analysis**, that based on the main characteristics of the penetration testing process, identifies a set of interpretive elements to enhance comprehension.
- **An interpretive mapping** that organizes the interpretive elements in the penetration testing process and automatically processes these elements into the visual narrative components necessary for display.
- **A visual summary system** that displays narrative components to summarize the penetration testing process.
- **A qualitative experiment** that assesses the effectiveness of the system in supporting the interpretability and comprehensibility of the penetration testing process.

II. RELATED WORK

This paper has researched the visual summary of the penetration testing process, we referred to the related research in the following fields: (1) penetration testing methodology and (2) network attack visualization.

A. PENETRATION TESTING METHODOLOGY

In the field of penetration testing, current research is mainly focused on penetration testing methods, involving methodological research that is more specific to the test object, such as tests for web servers, tests for wireless networks, and tests for web applications. Penetration testing methodologies are relatively mature in the field of penetration testing. Pete Herzong created the Open Source Security Testing Methodology Manual (OSSTMM) [10], which includes five types of detailed penetration testing cases and provides an index standard for evaluating security testing results. The Information Systems Security Assessment Framework (ISSAF) [11] is an open-source security testing and security analysis framework that determines the logical sequence of security assessment and establishes a comprehensive set of security testing rules and procedures. The Penetration Testing Execution Standard (PTES) [12] was jointly initiated by several enterprise technical experts in the field of network security. In this methodology, the basic standards required for penetration testing are established, the basic procedures and methods followed for penetration testing are specified, and the standardization of penetration testing procedures is completed.

The OSSTMM and ISSAF focus on the methods and guidelines for security assessment and vulnerability analysis. In contrast, the PTES has more standardized specifications on the execution process and methods of penetration testing. It has established a baseline of basic criteria for penetration testing, which is a standard commonly used in the field of penetration testing technology. Therefore, in the design, PTVis follows the PTES and conducts a visual construction of the penetration testing process based on three stages: information gathering, the execution of attacks, and the generation of results.

B. NETWORK ATTACK VISUALIZATION

Network attack visualization belongs to the field of network security visual analysis [13]. Interactive visual interfaces improve the ability of network personnel to perceive, analyse, and comprehend network security issues [14]. In terms of anomaly detection, Yang Shi *et al.* [15] proposed a radial visualization method for intrusion detection system (IDS) alarm log data using celestial bodies as a metaphor, solving the problem of visual clutter in radial visualization and the

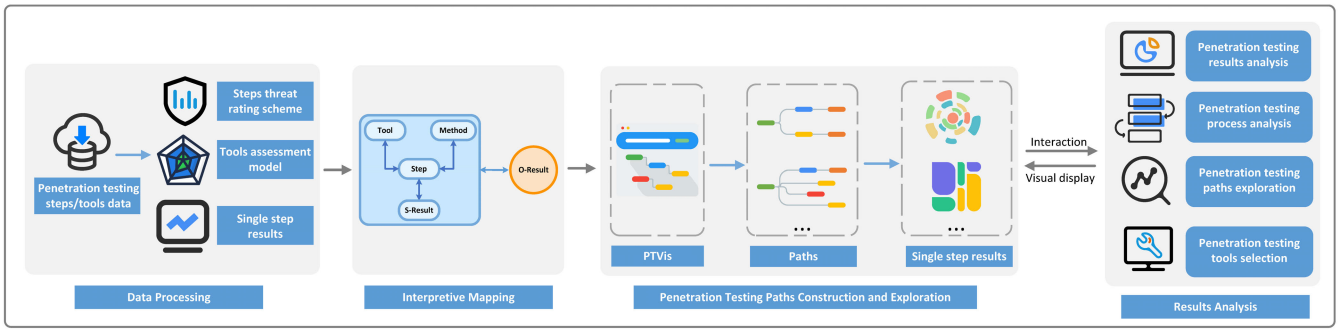


FIGURE 1. Analysis flowchart of PTVis. Through data processing, penetration testing-interpretive mapping, and visual presentation and interaction of the penetration testing paths summarize the process construction and results analysis of penetration testing.

dynamic aggregation of similar hosts and correlation between hosts. Angelini *et al.* [16] proposed a visual analysis solution for dynamically detecting network propagation vulnerabilities that can help in quickly comprehending the network status and providing optimal patch sequences based on node vulnerabilities. Ball *et al.* [17] designed VISUAL, a visualization system that shows the communication status of the internal and external network hosts. Based on the number of connections and the size of the external network host block, some unusually active internal and external network hosts can be found intuitively. In network monitoring, Mansmann *et al.* [18] used treemaps to represent the hierarchical characteristics of IP addresses, and users can freely view the hierarchical summary or detailed information through interaction. McPherson *et al.* [19] provided an interactive method of area selection and magnification observation to prevent the interaction of overly dense data points. Considering the different levels of importance of port number sections, less important port numbers were represented by smaller graphical elements. FlowScan [20] used a stack chart to visualize the temporal changes in traffic on a campus. In terms of identify correlations, Lakkaraju *et al.* [21] designed a multi-view tool, NVisionIP, that combined host monitoring and port monitoring with a global overview and local analysis through a 3-level interactive view. IDS Radar [22] explored the possible correlation between botnet infection events and file theft events in an enterprise network. Avisa [23] used radar charts to analyse the multi-step attack process of complex network intrusions. For network security situation assessment, Mckenna *et al.* [24] designed a network security dashboard to help network analysts identify and summarize patterns in data. The above work is aimed at the visual analysis of network security result data, and it focuses on helping researchers analyse large-scale network security data.

The visual analysis of penetration testing activities is a research direction in the visualization of network attacks, and there is currently little research in this field. In terms of penetration testing event analysis, Gregory *et al.* [6] researched the visual fingerprint information left by various popular penetration testing tools to help analysts better understand the specific methods used by attackers and the identifiable characteristics of the tools. Ankit *et al.* [7] designed a prototype workspace to help visualize the workflow of analysts when analysing network penetration events and to improve the efficiency of analysts in analysing network penetration events. ROPMate [8] provides a semantic-based visual analysis solution, that helps cyber red teams filter and link to form complete vulnerabilities by filtering penetration testing tools. In terms of penetration testing processes and methods, Yuen *et al.* [9] designed a CRT practice analysis visualization system based on the cyber red team context to improve the network situation analysis capabilities of analysts. Walton *et al.* [25] proposed a visual analysis cycle to support the continuous development of the network security model during deployment by visualizing the execution process of penetration testing and analysing network security threats. Briesemeister [26] analysed the process of network traversal attacks against digital control systems using tree diagrams. Most of these efforts focus on analysing a certain stage or aspect of penetration testing activities.

However, unlike the above work, PTVis focuses on analysing the processes, methods, and main characteristics of penetration testing. PTVis displays the penetration testing process in different ways from a global perspective. It aims to improve the interpretability and comprehensibility of penetration testing, enhance the network situation awareness of decision-makers, and facilitate the analysis of penetration testing data.

III. DATA AND PREPROCESSING

Through discussions with experts in visual analysis and penetration testing, data requirements that can summarize the penetration testing process were determined:

- The data are used to explore penetration testing methods and to represent different technical pipelines (labelled as paths) for penetration testing activities. Each type of penetration testing method consists of different penetration testing steps, and the implementation of steps depends on the penetration testing tools. The users of PTVis include non-professionals in an organization, thus the experts recommend reducing manual intervention and helping users explore penetration testing methods by the auxiliary decision. Therefore, the construction of the penetration testing path requires penetration testing step

TABLE 1. Penetration testing step data.

Id	*Source*	*Type*	*Target*	*Type*
P1	Directory-scanning	info	Backup-file-leak	info
P2	Directory-scanning	info	Source-code-leak	info
P3	Backup-file-leak	info	Backup-file-leak-utilization	attack
P2	Source-code-leak	info	Source-code-leak-utilization	attack
P3	Port-scan	info	Port-unauthorized-access	attack
P4	Port-scan	info	Port-blast	attack

data, penetration testing tool data, and tool evaluation data.

- The data are used to represent the single-step results and the overall results in the penetration testing process. Analysing the results of penetration testing can guide penetration testing activities and help users to better understand the effectiveness and necessity of penetration testing. Therefore, the data of the step threat level and the single-step result data are needed to display the penetration testing results.

The single-step result data can be directly mapped to the view, and other types of data need to be processed.

A. PENETRATION TESTING STEP AND TOOL DATA

The penetration testing step and tool data recorded were obtained through investigation. Through a survey of 4 teachers and 20 students in a university information security laboratory, 11 penetration testing methods and 64 penetration testing tools were recorded. The penetration testing methods include port blast, SQL injection, and cross-site scripting (XSS) attacks, etc.

Based on the penetration testing process from ''information gathering'' to the ''execution of attacks'', 11 recorded penetration testing methods are divided into steps to establish the correspondence between steps and tools. Each penetration testing step corresponds to 4 different penetration testing tools. The penetration testing step data are saved as shown in table 1 (only part of the data is shown). In PTVis, users choose different penetration testing steps to construct penetration testing paths in an incremental way. *Source* and *Target* indicate the execution order of the penetration testing steps, and *Id* identifies different penetration testing paths to indicate different methods. Different types of penetration testing steps are distinguished by the value of *Type*. ''Info'' means ''execution information gathering'', and ''attack'' means an ''execution penetration attack''.

B. PENETRATION TESTING TOOL EVALUATION DATA

Penetration testing tool evaluation is based on international software quality standards to quantify the performance and quality of tools [27]. ISO/IEC 25010:2011 [28] is an international standard for evaluating software quality promulgated by ISO/IEC in March 2011. It is an improvement to the ISO/IEC 9126:2001 software quality standards. ISO/IEC 25010:2011 is commonly used in evaluation applications [29], portal websites [30] and cloud computing systems [31]. The evaluation of the quality dimension of the application includes 8 quality characteristics: *functionality*, *safety*, *compatibility*, *reliability*, *usability*, *efficiency*, *maintainability* and *portability*. In general, after adding some specific evaluation characteristics or sub-features to the evaluation items, the application of international quality evaluation standards is achieved by calculating the expert score [32]. For example, Zeiss *et al.* [33] added *reliability* to apply ISO/IEC 9126:2001 to standardize testing. When evaluating the quality of business-to-business (B2B) applications, the sub-feature level was determined by adding *traceability*, *availability*, *customizability*, and *navigability* [34]. This paper improves the evaluation method proposed in reference [35] by adding the new features (*safety*, *compatibility*) proposed by ISO/IEC 25010:2011. Additionally, we select 20 network security practitioners (the average working period is 5 years, of which the minimum working period is 3 years) to perform a quality evaluation for each penetration testing tool.

In PTVis, the evaluation of penetration testing tools includes the overall quality evaluation of the tools and the evaluation of individual quality indicators of the tools. Set F_i ($i = a, b, c, \ldots, h$) to indicate *functionality*, *safety*, *compatibility*, *reliability*, *usability*, *efficiency*, *maintainability* and *portability*, and the score range is 1-10. The weight coefficient of the eight indicators is W_j ($j = 1, 2, 3, \ldots, 8$), the overall quality evaluation result of the tools is R_s, and the quality evaluation result of the single indicator of the tools is P_s. The formula is as follows:

$$R_s = \sum F_i \cdot W_j \tag{1}$$

$$P_s = \lfloor F_i \rfloor \tag{2}$$

According to the evaluation criteria of the 38 sub-features of the 8 quality characteristics in the ISO/IEC 25010 standards, the weights are determined by the experience of 20 selected network security practitioners, $W_1 = 0.2$, $W_2 = 0.1$, $W_3 = 0.05$, $W_4 = 0.2$, $W_5 = 0.1$, $W_6 = 0.2$, $W_7 = 0.05$, $W_8 = 0.1$ (the sum of the weight coefficients is equal to 1). The average value of the evaluation results of the 20 network security practitioners is calculated to obtain the evaluation data of the penetration testing tools.

C. THREAT LEVEL DATA FOR THE PENETRATION TESTING STEPS

The penetration testing step threat level refers to the threat brought by the execution of a single step in a penetration testing method. The Open Web Application Security

Project (OWASP) is a non-profit global security organization that regularly launches the Top 10 Project (top ten security hazard prevention codes) to enhance the public's awareness of application security. PTVis designs a penetration testing step threat rating scheme based on the ''OWASP risk ranking method'' in the 2017 OWASP Top 10 Project [36]. The threat level calculation is performed based on four aspects: availability *(A)*, universality *(U)*, detectability *(D)*, and technicality *(T)*. The formula is as follows:

$$Th = \frac{A + U + D}{3} \cdot T \quad A, U, D, T \in [1, 10] \tag{3}$$

Based on the calculation results, the threat level *(Th)* of the penetration testing steps is quantified into five levels, as shown below:

$$Th \in \begin{cases} [1.0, 20.0) & lowest \\ [20.0, 40.0) & low \\ [40.0, 60.0) & medium \\ [60.0, 80.0) & high \\ [80.0, 100.0] & highest \end{cases} \tag{4}$$

IV. CONSTRUCTION OF PENETRATION TESTING-INTERPRETIVE MAPPING

PTVis aims to design visual summaries based on visual narrative and the auxiliary decision to improve the interpretability and comprehensibility of the penetration testing process. To that end, we make design decisions based on the display requirements of penetration testing data. Then, we divide the penetration testing process into different stages and analyse the prominent elements in each stage. Based on the structure of these elements, penetration testing-interpretative mapping is constructed to drive the visual design [37].

A. INTERPRETIVE ELEMENTS SELECTION DURING PENETRATION TESTING

First, the penetration testing process is divided into different stages. The PTES divides the penetration testing process into seven stages: pre-engagement interactions, intelligence gathering, threat modelling, vulnerability analysis, exploitation, post- exploitation, and reporting [12]. The typical kill-chain of the attacker divides the penetration attack process into three phases: reconnaissance, the execution of attacks, and exploitation [38]. To better assist the analysis of network attacks, PTVis combines the PTES and the typical attacker kill-chain from the perspective of an attacker. The penetration testing process is divided into the preparatory phase → execution phase → analysis phase.

PTVis aims to design a lightweight visual summary. We attempt to identify some of the prominent elements in these three stages, which can effectively characterize the penetration testing process and assist the user's comprehension. To determine the primary interpretive elements of the penetration testing process, 10 experts with more than 3 years of penetration testing experience (5 network security engineers, 2 university professors, and 3 experts who hold a doctorate in information security) were invited to participate in a sample survey. Using the Delphi method, we conducted three questionnaire survey rounds with these 10 experts. The process is as follows:

1) THE FIRST ROUND OF THE DELPHI STUDY

The first round of the Delphi survey sent relevant materials and questionnaires to all 10 experts. These materials included the research content, research objectives, and data types of our work. The content of the questionnaire is as follows:

- What stages are necessary in the actual implementation process of penetration testing?
- What are the key steps in the actual implementation of penetration testing?
- What information is the focus of attention in the analysis of the results of the penetration testing?

All experts were required to list the interpretive elements of the penetration testing process based on the content of the questionnaire and referring to relevant materials. When ten experts completed the first round of the questionnaire survey, we sorted out the elements proposed by the experts, merged similar elements, and used the results in the next round of the Delphi survey.

2) THE SECOND ROUND OF THE DELPHI STUDY

The purpose of the second round of the Delphi survey is to establish a consensus among the experts on the importance of the interpretive elements of each penetration testing process. The experts were asked to rate the elements summarized in the first round of the surveys (from 1 = the least important to 5 = the most important). The results are shown in tables 2, 3 and 4.

The average score of all elements in the preparatory phase is 3.64 and that in the execution phase is 3.95. The final stage, the analysis phase, has an average score of 3.4. We consider the three average scores to be the cut-off points, and only the interpretive elements of the average scores that are greater than the cut-off point of the same stage will be sorted again in the next study. In the Delphi study, there is no universally agreed-upon cut-off point. Although a cut-off point is arbitrary, the average score is still considered appropriate measure to use as the cut-off point [39], [40].

After the second round of the Delphi questionnaire survey, five interpretive elements of the penetration testing process met the importance evaluation threshold: attack execution, information gathering, vulnerability analysis, vulnerability modelling, and single-step result analysis. These elements are evaluated again in the next round of the study.

3) THE THIRD ROUND OF THE DELPHI STUDY

In the third round of the Delphi surveys, the experts were asked to reconsider their scores on interpretive elements based on the results of the second round. Most experts reconsidered their evaluations and adjusted their ratings. In the preparatory phase, the experts believed that vulnerability modelling included vulnerability analysis; thus, the

TABLE 2. Results of the second-round Delphi study—primary interpretive elements of preparatory phase.

Rank	*Primary interpretive element*	*Std Dev*	*Median*	*Mode*	*Mean*
1	Information gathering	0.72	4	4	4
2	Vulnerability analysis	0.67	3.5	4	3.8
3	Vulnerability modelling	0.75	4	4	3.7
4	Risk analysis	0.82	3	3	3.4
5	Pre-engagement interactions	0.79	3	3	3.3

TABLE 3. Results of the second-round Delphi study—primary interpretive elements of execution phase.

Rank	*Primary interpretive element*	*Std Dev*	*Median*	*Mode*	*Mean*
1	Attack execution	0.69	4.5	4	4.4
2	Post-penetration attacks	0.85	3	3 and 4	3.5

TABLE 4. Results of the second-round Delphi study—primary interpretive elements of analysis phase.

Rank	*Primary interpretive element*	*Std Dev*	*Median*	*Mode*	*Mean*
1	Single-step result analysis	0.63	3.5	4	3.6
2	Reporting	0.65	4	3	3.4
3	Information management	0.74	3	3	3.2

vulnerability analysis score fell below the importance evaluation cut-off point (3.64). In the analysis phase, the experts took into account the important role of single-step result analysis in the execution of penetration testing and increased their scores.

Based on the results of the three rounds of the Delphi surveys, the four primary interpretive elements of penetration testing are defined as follows:

① *Information gathering*. This is carried out through direct access, scanning, and other methods to collect information such as domain names, fingerprint identification, directories, and files of penetration testing targets [41], as well as behavioural patterns and operating mechanisms;

② *Vulnerability modelling*. This is carried out by scanning the ports and hosts of the penetration testing target, analysing the possible security vulnerabilities and weaknesses in the target system, and determining the appropriate penetration testing methods;

③ *Attack execution*. This is carried out by using the vulnerability of the target system to execute the attack methods based on the vulnerability modelling results;

④ *Single-step result analysis*. With the execution of the penetration testing, the result information generated at each step is analysed to determine whether to continue the testing.

PTVis takes into account the interpretive elements of the penetration testing process from a cognitive perspective. At this level, ''comprehension'' can be defined as an improvement in the awareness of a specific process through the exploration of the ''user + scenario'' [42]. To improve recognition and deepen the comprehension of the penetration testing process, we need not only the construction of penetration testing paths to explore the analysis results but also the penetration testing results to retrospectively analyse the path construction process. Interpretive elements ①-④ of the penetration testing process specify the direction from path construction to result analysis. For the retrospective analysis, PTVis specifies two additional primary interpretive elements:

⑤ *Time consumption statistics of the steps*, which record the time consumption of each penetration testing step, trace back the construction phase of the penetration testing process and analyse whether the penetration testing methods or tools corresponding to the steps are efficient;

⑥ *Threat level calculation of the steps*, which calculates the threat level of each penetration testing step and analyses the penetration testing methods with a high threat level.

The above six interpretive elements are used to summarize the penetration testing process and to promote comprehension and analysis, as shown in Figure 2.

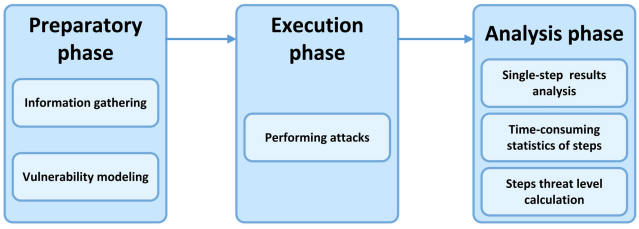


FIGURE 2. Primary interpretive elements in the three stages to summarize the penetration testing process.

B. PENETRATION TESTING-INTERPRETIVE MAPPING

The next step consists of structuring the six defined penetration testing elements to visually characterize and summarize the penetration testing process. Based on their definitions, the six interpretive elements are mapped to a set of narrative components that exist within all penetration testing activities: The *Step*, *Tool*, *Method*, *S-Result*, and *O-Result*.

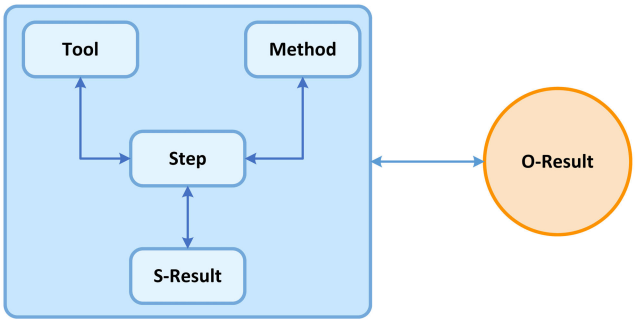


FIGURE 3. Penetration testing-interpretive mapping. ***Step*** **represents all the penetration testing steps in information gathering, vulnerability modelling, and attack execution;** ***Tool*** **represents all penetration testing tools in information gathering and attack execution;** ***Method*** **represents vulnerability modelling and all penetration testing methods in attack execution;** ***S-Result*** **represents the single-step results;** ***O-Result*** **represents the overall exploration results.**

The visual narrative components and their associations are shown in Figure 3.

Penetration testing-interpretive mapping is constructed with the *Step* as the main component. The *Step*, *Tool*, *Method*, and *S-Result* construct the interpretive part of the penetration testing process, and *O-Result* constructs the interpretive part of the penetration testing result.

1) THE INTERPRETATION OF THE PENETRATION TESTING PROCESS

The visual components are interrelated, and the penetration testing tool is selected by the correspondence between the *Step* and the *Tool*. The determination of the tool promotes the progress of penetration testing. All penetration testing methods are composed of several steps. The *Step* is a collection of all penetration testing steps in the *Method*. The *S-Result* is a quantitative representation of the progress of penetration testing. As the penetration testing progresses, the single-step results are generated based on different penetration testing step selection tools. Therefore, when the single-step results are analysed, it is necessary to select an appropriate penetration testing method or to adjust the current method to gradually explore the construction process of penetration testing.

2) THE INTERPRETATION OF PENETRATION TESTING RESULTS

The *O-Result* includes the single-step results, time-consuming steps, and step threat levels. That is, after three different types of results are aggregated, the results of penetration testing are used to retrospectively analyse the construction process of penetration testing paths.

The data in PTVis are relational data [43], which correspond to the progress of penetration testing. The construction of penetration testing-interpretive mapping is based on the execution process of the penetration testing steps. The selected visual metaphor should emphasize the connection between the interpretive elements and the visual components. Therefore, selecting tree structure visualization as the basic technology can effectively show the progress of penetration testing and the relationship between the elements. The next section describes the visual mapping of PTVis.

V. PTVis DESIGN

Based on the interpretive mapping and sample survey, the visual design requirements of PTVis are as follows:

DR1 Provide a visual display of the interpretive elements of the penetration testing process. By reducing the required cognitive knowledge, recognizable, explorable, and simplified visual effects make it easier for users to comprehend penetration testing.

DR2 Organize the interpretive elements of the penetration testing process to help analyse the main characteristics of penetration testing. To visually summarize the penetration testing process, it is necessary to associate the primary elements as components rather than as independent events. Emphasizing the connection of the various stages of penetration testing, from the details to the overall analysis, is helpful in comprehending the penetration testing process from multiple angles and in depth.

DR3 Enable users to the customized exploration of the penetration testing process and to analyse the penetration testing results. Users may choose the penetration testing path of interest to explore or compare multiple paths simultaneously; thus, they can analyse the single-step results or the overall exploration results. Users should be provided with options to select and add/remove penetration testing paths and to modify other style settings.

A. INTERPRETIVE VISUALIZATION OF THE PENETRATION TESTING PROCESS (DR1 AND DR2)

The visual design of PTVis revolves around five visual components. The *Step*, *Tool*, *Method*, and *S-Result* construct the process-interpretive visualization, that is, the penetration testing path visualization construction part. The *O-Result* constructs the result-interpretive visualization, that is, the penetration testing result visualization part. The five visual components are interrelated so that the entire penetration testing progress can be mapped with the progress of the penetration testing steps.

Based on the suggestions of related experts, PTVis uses black as the background colour of the system, as black can reduce user eye fatigue [44] and improve the discernibility of other elements in the system [45]. This design also ensures that the visualized view can be easily recognized by the user. In PTVis, the visual elements use brighter and more saturated colours to display data information completely in a purposeful manner. Through comparison with the system background, users can perceive the focus information in the visualized view [46], and the user's attention is focused on exploring the penetration testing process.

1) VISUALIZATION OF PENETRATION TESTING PATHS

Penetration testing is composed of different stages, and analysing the correlations between penetration testing steps is of great importance in analysing the main characteristics of penetration testing. The visualization of penetration testing paths constitutes the main view of the interface for the penetration testing process. The interface components are shown in Figure 4A-E. The purposes of these components are as follows:

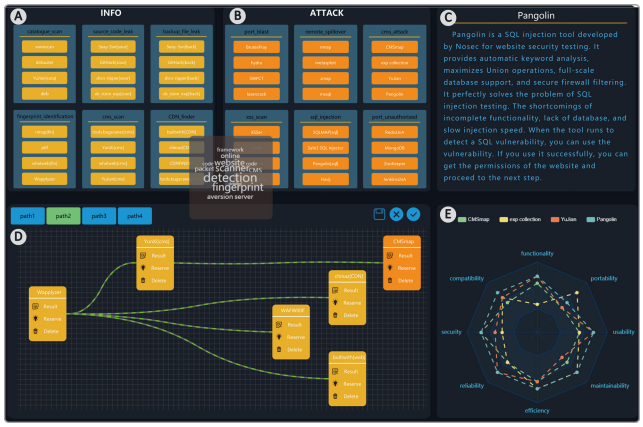


FIGURE 4. Interface of the penetration testing process. (A\B) The step-toolbar displays the steps and tools of different penetration testing methods; (C) the information box displays the introduction to the steps and tools and the view of the single-step results; (D) the working area displays the penetration testing paths, while users can choose different penetration testing steps and tools to construct penetration testing paths; and (E) the tool evaluation view displays the results of the penetration testing tool evaluation.

A\B Step-toolbar: Displays the classification and correspondence of the penetration testing steps and tools;

C Information box: Shows the introduction to the penetration testing steps and tools and the view of the single-step results;

D Working area: Used to visually construct the penetration testing paths;

E Tool evaluation view: Displays the evaluation results of the penetration testing tools.

a: PENETRATION TESTING STEPS AND TOOLS

There are many steps and tools corresponding to different penetration testing methods. Thus, showing the penetration testing steps and tools in a clear, concise way makes it convenient for users to make appropriate choices. In accordance with the correspondence between the penetration testing steps and tools, all steps and tools are displayed in the form of a step-toolbar, as shown in Figure 4-A/B. The encoding with different colours can distinguish the various purposes of the penetration testing steps: information gathering (info) or penetration attack execution (attack). To help users quickly understand the purpose of different steps and the use of different tools, the introduction to the penetration testing steps and tools is displayed in the form of a ''text + word cloud'', which is shown in Figure 4C. Although word clouds have drawbacks [47], word clouds provide a simple and effective way to visually convey the most frequent words in a text [48], [49]. We extract the keywords of the word clouds from the text information of the penetration testing steps and tools, and the size of the keywords indicates their importance. In PTVis, the number of keywords in the word clouds is usually small and readable, which can help users quickly understand different penetration testing steps and tools. Therefore, word clouds constitute a feasible visualization method. Word clouds are hidden by default and are shown by hovering over the toolbar with the mouse.

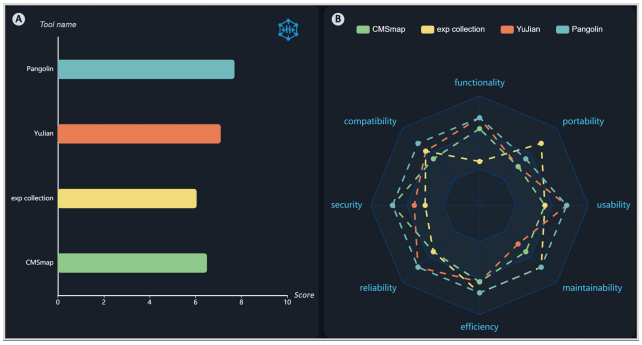


FIGURE 5. Quality evaluation results of the penetration testing tools. Four types of penetration testing tools are assessed for the *cms-attack* penetration testing step: (A) the overall quality evaluation of the tools; and (B) the individual quality evaluation of the tools.

In the construction of the penetration testing paths, choosing suitable penetration testing tools can meet the needs of different steps or stages. To visually compare the performance of penetration testing tools, we encode the overall quality evaluation results and the individual quality evaluation results of the tools, as shown in Figure 5. The horizontal histogram encodes the overall quality evaluation results of the four tools of the same penetration testing step, and it visually shows the comprehensive performance of the different tools. Users can switch to a multi-dimensional radar chart by clicking on it. This chart encodes the eight quality indicators of the tool (e.g., *functionality* and *usability*). Thus, the penetration testing tools can be compared from different performance perspectives to help users make accurate choices under different requirements.

b: PENETRATION TESTING PATH

To assist the user's comprehension, we metaphorically refer to penetration testing methods as technical pipelines and label them as paths. Users construct penetration testing paths based on the correlations among the penetration testing steps to explore different penetration testing methods. PTVis designs penetration testing paths based on tree diagrams. Compared with other visualization approaches that support relational data (e.g., treemaps, sunburst, packed circles), tree diagrams focus on the connections between nodes and visually conform to the metaphor of penetration testing paths. Moreover, tree diagrams have excellent interactivity, and they can clearly show the dynamic changes in penetration testing and the thinking mode of users when exploring penetration testing

methods. PTVis designs penetration testing paths based on a tree diagram. ''Process decision thinking'' is introduced based on the tree diagram, and the ''process decision tree diagram'' is innovatively designed. The process decision tree diagram combines the characteristics of the common tree diagram, which can show the relational structure, with the auxiliary decision, which equips it with the characteristics of traceability and dynamic management and completes the interpretive construction of the penetration testing path.

Each penetration testing method consists of different penetration testing steps. Meanwhile, the implementation of these penetration testing steps depends on penetration testing tools. Based on the correspondence between the steps and tools, the penetration testing tools are selected as nodes in the process decision tree diagram (labelled as tool nodes) to dynamically construct the penetration testing path. Each tool node corresponds to a penetration testing step and is encoded with the same colour as the step to distinguish the purposes of the tool node: information gathering or penetration attack execution. The initial tool node will automatically create a uniquely identified penetration testing path working area. In these working areas, the connection between the tool nodes indicates the penetration testing path, as shown in Figure 4D. The user builds a penetration testing path incrementally by selecting the penetration testing tool in the step-toolbar. To emphasize the connection between the penetration testing steps and the single-step results, the tool nodes add some interaction methods, for example, viewing results and deleting nodes. Users can view the results of the current step in the information box, as shown in Figure 6C (this is the result of the port scanning of the target website; it helps users identify various port vulnerabilities quickly by providing the available type clustering, port status statistics, and risk calculation). To better support exploration, PTVis provides auxiliary decisions for subsequent steps or methods based on the penetration testing steps selected by the user, as shown in Figure 6B. The user analyse the execution status of the current step and the vulnerability of the penetration testing object via the single-step result view and determines whether it is necessary to continue to perform subsequent penetration testing steps or to change the current penetration testing method.

The flow direction of the lines between tool nodes shows the execution sequence and data flow of the penetration testing steps. A tool node may have multiple lines, indicating that different penetration testing methods include the same steps, such as the first node in Figure 4D. To characterize the complete process of a type of penetration testing method, PTVis constructs a penetration testing path in the following manner: ''create a node → view result + auxiliary decision → create a node. . .''.

2) VISUAL CONSTRUCTION OF PENETRATION TESTING RESULTS

To improve recognition and deepen the comprehension of the penetration testing process, we need not only the construction

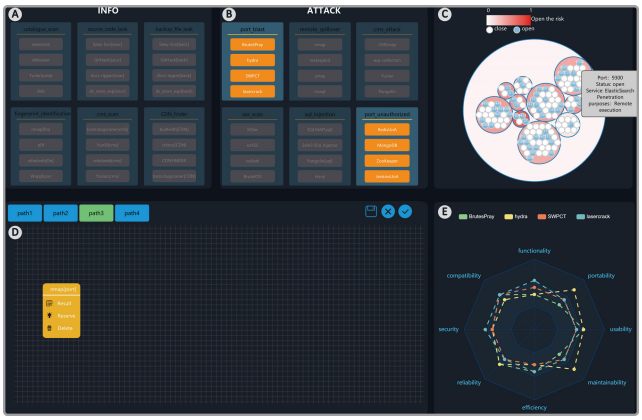


FIGURE 6. Auxiliary decision and single-step result view. (B) provides the auxiliary decision for the subsequent steps or methods of the *port scan* step; and (C) shows the result view of the tool node corresponding to the *port scan* step.

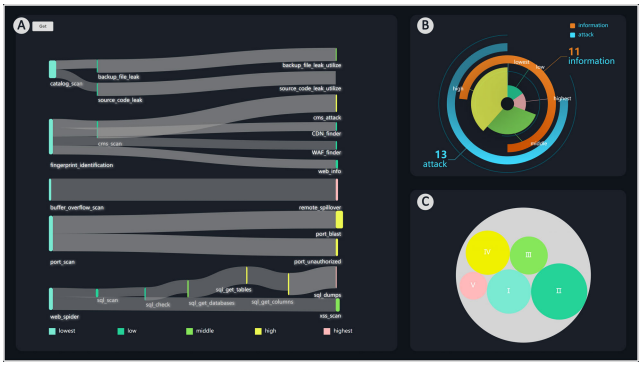


FIGURE 7. Interface of the penetration testing results. (A) The results area displays the penetration testing result Sankey diagram, which helps users backtrack the construction process of the penetration testing paths; (B) the node statistics view, in which the outer circle shows the type and number of step nodes; by clicking on the outer circle, the inner pie chart shows the threat level of step nodes of the same category; and (C) the vulnerability statistics view, which shows the number of vulnerabilities and threat levels in the penetration testing results.

of penetration testing paths to explore the analysis results but also the penetration testing results to retrospectively analyse the path construction process. PTVis designs a penetration testing results interface based on the Sankey diagram. Sankey diagrams are a specific type of flow chart that have the advantage of supporting relational data and network data at the same time and having good scalability. Three types of attributes, i.e., the process time, step threat levels, and single-step results, are added in the Sankey diagram. Meanwhile, the path between the step nodes is mapped to the ''data channel'' to demonstrate the data traffic among the different steps of the penetration testing methods. We evolve the Sankey diagram into the ''penetration testing result Sankey diagram'' so that users can carry out a customized exploration of the penetration testing result display. The penetration testing result Sankey diagram constitutes the main view of the penetration testing result display interface, and the interface components are shown in Figure 7A-C. The purposes of these components are as follows:

A Results area: Displays the penetration testing result Sankey diagram;

B Node statistics view: Used to count the classification and quantity of nodes in the penetration testing result Sankey diagram;

C Vulnerability statistics view: Used to count the number of vulnerabilities and threat levels in the penetration testing results.

Based on the corresponding relationship between the penetration testing steps and tools, the tool nodes in the penetration testing path constructed by users are bound to the corresponding steps. The nodes in the penetration testing result Sankey diagram represent the penetration testing steps, as shown in Figure 8A.

FIGURE 8. Penetration testing result Sankey diagram. (A) Penetration testing paths that users construct autonomously; (B) the mask layer, which can display the single-step result view and related data of the step nodes.

The higher the height of one step node is, the greater the number of sub-nodes that represent this step node. This means that the results and data of this step node can be used by multiple subsequent steps, further demonstrating that this penetration testing step is key to the success of n (number of child nodes) penetration testing methods. The lines between step nodes are mapped as ''data channels''. The wider the connection between two nodes is, the greater the amount of data that are available from the node in the previous step. Therefore, more attention should be paid to the previous penetration testing step.

To help analyse and identify the threat levels of penetration testing steps and methods, the step nodes are colour-coded with a green-yellow-red gradient colour transition, corresponding to five penetration testing step threat levels (lowest, low, medium, high, highest). Based on the penetration testing paths constructed by the user, the time consumption data of the tool nodes and the single-step result data are saved. Then, the time consumption situation is shown by the width of the step node. The wider the node is, the longer the penetration testing step takes, or the lower the efficiency of the selected penetration testing tool, as shown in Figure 8A. At the same time, the Focus + Context method [50] can display the focus information through the superimposed layer, thereby constructing a contextual situation to reduce the user's cognitive load. To emphasize the dynamic changes when the user constructs penetration testing paths and to help the user backtrack the penetration testing path construction process, we add a mask layer in the penetration testing result Sankey diagram to display the single-step result view and related data. By clicking on any step node, the interface displays the single-step result view and related data of the node in the mask layer, as shown in Figure 8B. This feature helps users review the penetration testing path construction process and summarize the characteristics of different penetration testing methods.

B. INTERACTIVE DESIGN (DR3)

PTVis provides a variety of interaction methods for gaining better user comprehension and more support for exploration.

1) INTERACTIVE DESIGN FOR CONSTRUCTING PENETRATION TESTING PATHS IN PARALLEL

PTVis can construct multiple penetration testing paths simultaneously and provide interactive methods such as selection and switching. Users can choose different penetration testing tools to create multiple working areas for penetration testing path construction. Each working area has a unique identifier, and users can click on the identifier to enter the working area. Combined with the subsequent penetration testing steps and methods provided by the auxiliary decision, the user analyses the execution of the current step via the single-step result view to determine whether it is necessary to continue to perform the subsequent steps or to change the current method. The penetration testing methods corresponding to the working area are incrementally explored in the following manner: ''create a node → view results + auxiliary decision → create a node. . .''. Moreover, the user can switch different working areas at any time and construct multiple penetration testing paths to efficiently analyse different penetration testing methods.

2) INTERACTIVE DESIGN FOR ANALYSING THE CONSTRUCTION RESULTS OF PENETRATION TESTING PATH

To help users analyse the dynamic changes in the penetration testing process, PTVis provides a penetration testing path construction method that combines penetration testing process construction and result analysis. Users can see the single-step result view of any node in the penetration testing path diagram and analyse the correlations between the penetration testing steps. Through the combination of steps and single-step results, the user determines whether the penetration testing path constructed is successful and whether the path needs to be saved to the penetration testing results interface for retrospective analysis.

3) INTERACTIVE DESIGN FOR BACKTRACKING PENETRATION TESTING RESULTS-PROCESS

PTVis builds a connection between the ''penetration testing process construction interface'' and the ''penetration testing result display interface'' to help users clearly analyse the correlations between the penetration testing process and results. In the penetration testing result Sankey diagram, users can see the result view and corresponding data of any step node in the

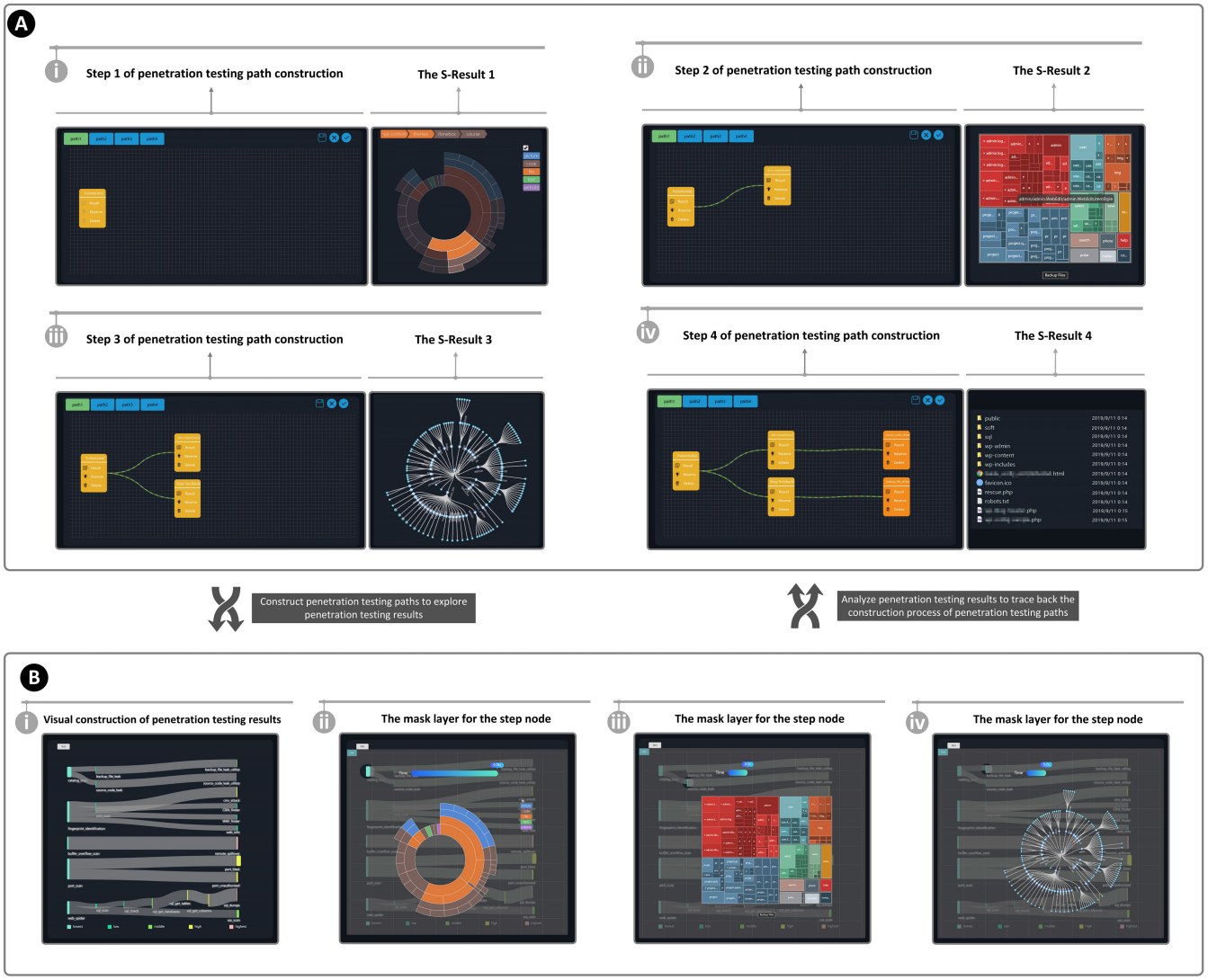


FIGURE 9. Penetration testing process of the target site source code and backup file exploration and utilization. (A) displays a visual construction of the penetration testing path, where users explore the extraction and utilization process of the target website source code and backup files through the auxiliary decision; and (B) represents the visualization of the penetration results, which displays the completed penetration testing paths, single-step result view and various data.

mask layer, select the penetration testing method of interest, and then backtrack to the working area where the path was constructed. The influence of the penetration testing results is analysed based on the selection of tools and the modification of methods.

C. USAGE SCENARIO

Penetration testing is typically used for website security assessment. By simulating an adversarial network attack, such testing can conduct a comprehensive detection of the target website, discover and repair website vulnerabilities, and thus prevent the occurrence of unsafe incidents such as website information leakage. Website vulnerabilities may cause the disclosure of internal IP information and user information. Moreover, website source code and backup files usually contain the directory structure and sensitive files of the website. Therefore, attackers can use the source code and backup files of the target website to obtain database information and other important system information, causing the core data of the website to leak. Based on the above theory and technology and following the penetration testing process from ''information gathering'' to ''attack execution'', the penetration testing process for the source code and backup files of the target website is selected to demonstrate the usage scenarios of PTVis.

1) CONSTRUCTION OF PENETRATION TESTING PATH

First, the exploration and construction of the penetration testing path are carried out. The text and word cloud help users understand the purpose of different penetration testing steps and tools. On this basis, users can compare the performance of different penetration testing tools from multiple

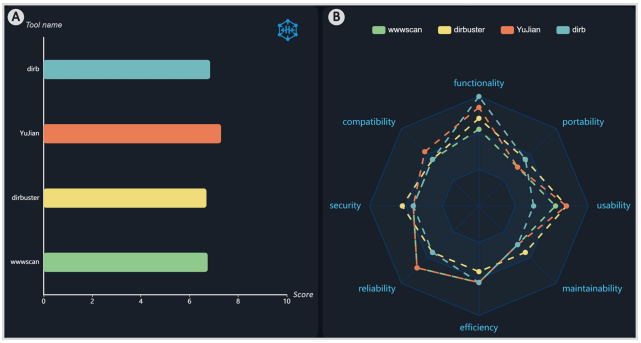


FIGURE 10. Evaluation results of four types of penetration testing tools corresponding to the *catalogue-scan* step. (A) The overall quality evaluation of tools; and (B) the individual quality evaluation of tools.

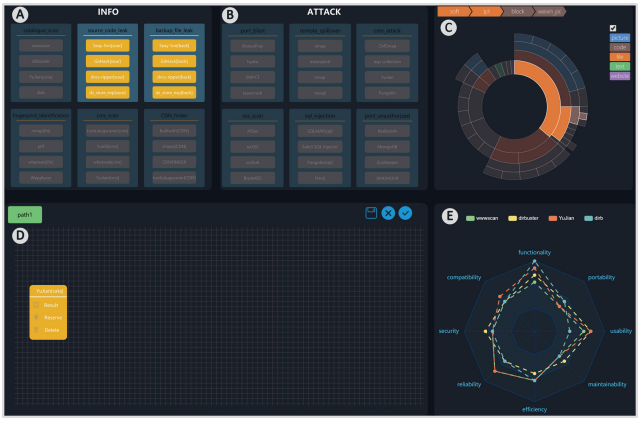


FIGURE 11. The subsequent penetration testing steps and methods of the *catalogue-scan* step are provided by the auxiliary decision, which can be used to analyse the selection of subsequent steps and methods based on the single-step result view of this step.

dimensions and choose a reasonable tool based on different needs. Based on the introduction of the penetration testing steps, performance comparisons of the four types of tools corresponding to the *catalogue-scan* step are performed, as shown in Figure 10. In terms of single tool performance, we focus on *functionality* and *usability*. Combined with the comprehensive performance of the tool, "*Yujian*" is selected to create the initial tool node and penetration testing path working area.

"*Path1*" is the unique identifier of one of the working areas that carries out penetration testing. By clicking on it, the single-step results of the initial node can be viewed, as shown in Figure 9A-(i). In this step, a directory scan of the target website is performed. It aims to complete the information gathering for the penetration testing target. In the field of penetration testing, the structure, file types, and data volume of the website directory usually need more attention in directory scan results. PTVis displays the results based on the sunburst model [51], [52]. The file types are encoded in different colours. The area of the fan-shaped block corresponds to the amount of file data, and an interactive way of displaying the path of the directory structure is employed. Users can analyse the directory structure and vulnerability of the target website based on the attacker's perspective. In the information gathering stage, the single-step results of the initial node contain a large amount of source code and backup file information about the target website.

The subsequent penetration testing steps and methods provided by the auxiliary decision can help users understand the purpose of information gathering and assist them in constructing a reasonable penetration testing path based on the information gathering results. As shown in Figure 11A, based on the subsequent penetration testing steps provided by PTVis, the source code and backup files of the target website can be extracted and utilized during the vulnerability modelling stage.

In the penetration attack execution stage, based on the comprehensive performance of the penetration testing tools, "*dvcs-ripper*" and "*Seay-Svn*" are selected to create tool nodes that extract and utilize the source code and backup files of the target website. In general, there are many complex files of website source code of various types. Penetration testing analysis concerns the structure and usage of source code. PTVis displays the source code files based on the treemap model [53], as shown in Figure 9A-(ii). Based on the different types of source code and the amount of information, different colours and proportions are used for encoding, and interactive operations that can download any source code files are added. In analysing a website's backup files, attention is typically paid to the path structure of the backup files, that is, the structure information of the website. To overcome the problem of excessively long backup file paths, PTVis displays the path structure of the backup files based on the radial tree model [46], as shown in Figure 9A-(iii). A hierarchical relationship between files is displayed through a tree structure, and an interactive method is added to download any backup file. As shown in Figure 9A-(iv), by extracting and using the website source code and backup files, the user can analyse the physical and logical structure of the target website and has the ability to detect the security defects of the source code and the structural defects of the website.

Simulating penetration testing scenarios helps users comprehend the importance of information gathering for penetration attacks, and it helps them explore the correlations among different stages of penetration testing. After the penetration testing path is constructed, the path is saved to the penetration testing result interface for retrospective analysis.

2) BACKTRACKING OF THE PENETRATION TESTING RESULTS

In the backtracking stage, based on the penetration testing result interface, the penetration testing path constructed is analysed. Through the penetration testing result Sankey diagram and the mask layer, the threat level of the step nodes in the path, the time consumption of the step and the single-step analysis results are shown in Figure 9B.

We backtrack the penetration testing process for the source code and backup files of the target website. In terms of

the characteristics of the step threat level, all steps are at the medium or low threat level, as shown in the first group paths of Figure 9B-(i). When constructing the backtracking path, other penetration testing methods can be selected for re-exploration to compare the threat levels of different methods. In terms of the time consumption of the steps, the directory scanning step is the most time-consuming step in the entire penetration test path, as shown in Figure 9B-(ii). The reason is that the step of penetration testing tool selection concerns only *functionality* and *usability*, rather than *efficiency*. Therefore, in a retrospective analysis, the choice of tools should be considered in terms of more characteristics. Regarding the single-step results, the result views of all the step nodes in the path are traced back by viewing the mask layer to explore whether there are missing details, and the characteristics of the penetration testing results are summarized, as shown in Figure 9B-(ii) to 9B-(iv). Retrospective analysis enhances the integrity of penetration testing. It enables users to view the entire penetration testing process from a global perspective and to analyse the correlations between different steps and stages.

Through the combination of path construction and result backtracking, users can explore more details of penetration testing. This promotes users' comprehension of the characteristics and security issues of different penetration testing methods.

VI. EVALUATION

To evaluate the effectiveness of PTVis, we conducted qualitative experiments on two groups (G1 and G2) and invited four experts for guidance (2 security engineers from network security companies and 2 professors in an information security lab at a university).

A. METHOD

In the field of penetration testing, the penetration testing process is usually summarized in the form of a text report. The user analyses penetration testing activities by viewing a text-based summary (including text narration and related pictures) to obtain relevant knowledge. Therefore, the evaluation of PTVis is divided into an objective evaluation and a subjective evaluation to compare PTVis and the text-based summary (this text-based summary is provided by a network security training company, and the details can be viewed in the supplementary material).

The objective evaluation focused on the participants' use of PTVis and the text-based summary to gain knowledge about the penetration testing process. The study consisted of two stages: (1) G1 viewed the penetration testing text summary, while G2 adopted PTVis to explore the penetration testing process. There were no other auxiliary tools. To prevent mutual interference, the two groups of participants conducted their evaluation in two rooms, with each having two external auditors. The maximum time at this stage was no more than 90 minutes. (2) G1 and G2 were required to complete a questionnaire test on the following day. To prevent teamwork, the participants were tested separately. Moreover, the test questionnaire was designed by experts (the details can be viewed in the supplementary material). The contents of the questionnaire included the principle, process, tool selection, and result analysis of the penetration testing. The question type and score were as follows: multiple-choice questions ($20 * 0.2$ points), fill-in-the-blank questions ($10 * 0.2$ points), short-answer questions ($4 * 0.5$ points), comprehensive analysis questions ($1 * 2$ points). The total score of the test questionnaire was 10 points. The maximum time at this stage is no more than 90 minutes. After the test questionnaire was completed, experts scored the test results based on the correct answers.

The subjective evaluation focused on user's psychological acceptability when using PTVis and the text-based summary. The study consisted of two stages: (1) the two groups exchanged experimental content. G1 utilized PTVis to explore the penetration testing process, while G2 viewed the penetration testing text summary. There was no other auxiliary tool. To prevent mutual interference, the two groups of participants conducted their evaluation in two rooms, with each having two external auditors. The maximum time at this stage was no more than 90 minutes. (2) We administered a short survey and questionnaire in which G1 and G2 rated how PTVis and the text-based summary affected their acquisition and comprehension of knowledge related to the penetration testing process using a 7-point Likert scale (adapted from [54]). Specifically, the questionnaire required ratings on how PTVis and the text-based summary influenced the following content:

- **Helpfulness**. This approach enhances comprehension and knowledge acquisition of the penetration testing process.
- **Interpretation**. This approach better characterizes the penetration testing process.
- **New Details**. Something not previously noticed is discovered.
- **Practicability**. The new details found will help in the next penetration testing.
- **Informativeness**. The essentials of penetration testing are demonstrated.
- **Readability**. The characterized content is clear and comprehensible.
- **Consistency**. The penetration testing process is presented in a logical and coherent form rather than in a fragmented form.

The score range was $[-3, 3]$, where -3 points means the participant preferred the text-based summary, 0 points means that there was no preference, and 3 points means that the participant preferred PTVis.

B. PARTICIPANTS

The experimental team recruited 28 participants (7 female) via the university email system, with an average age of 20.2 years (standard deviation $= 3.2$). The participants

TABLE 5. Participants statistics.

Major	*Number*	*Male/Female*
Computer Science and Technology	8	7/1
Software Engineering	7	5/2
Automation	2	2/0
Industrial Design	6	3/3
Communications Engineering	5	4/1

included university students and professionals in computer science and technology, software engineering, automation, industrial design, and communications engineering, as shown in table 5.

No participants had experience in penetration testing. They were randomly divided into two groups, G1 and G2, and all participated in the objective and subjective evaluation.

C. RESULTS AND ANALYSIS

In the objective evaluation, the participants used the text-based summary and PTVis to obtain knowledge about the penetration testing process. During the questionnaire test phase, most participants in G1 (using the text-based summary) had difficulty answering all of the questions, and the answers were not comprehensive enough to indicate that the participants recalled many details. Compared with G1, all 14 participants in G2 (using PTVis) were able to deepen their comprehension of the penetration testing process in an autonomous way and to acquire more knowledge, such as more detailed penetration testing methods and the data flow between steps. In statistics, the t-test is often used to determine whether the difference between the mean of two samples and the population represented by each is significant. Therefore, we performed a paired t-test on the scores of the two groups:

- Original hypothesis: The scores of G1 and G2 are not different;
- Alternative hypothesis: The scores of G1 and G2 are significantly different.

The results are shown in table 6. When the significance level is 0.05, the critical value of the rejection range of the two-tailed test is 2.1604, and the sample value of the test statistic t is 4.8575, and $p = 0.0003$. Since $t = 4.8575 > 2.164$ and $p = 0.0003 < \alpha = 0.05$, the original hypothesis is rejected and the alternative hypothesis is accepted. In the t-test, the smaller the *p-value* is, the smaller the probability of the original hypothesis. Because $p = 0.0003 < 0.001$, there is sufficient evidence to reject the original hypothesis, indicating that the scores of the two samples are significantly different. In addition, the score of G2 (using PTVis) is better than that of G1 (using text summary). In G1, the maximum and minimum scores are 7.2 and 4.1, respectively. In G2, the maximum and minimum scores are 8.4 and 5.5, respectively, indicating that the overall score interval of G2 is better than that of G1. The average is used to reflect the general level of the phenomenon. The average score of G2 is 6.46, while

TABLE 6. The t-test results of G1 and G2.

	G2	*G1*
Mean	6.46	5.34
Variance	0.9842	0.738
Observations	14	14
Pearson Correlation	0.5782	
Hypothesized Mean Difference	0	
df	13	
t Stat	4.8575	
P(T<=t) one-tail	0.0002	
t Critical one-tail	1.7709	
P(T<=t) two-tail	0.0003	
t Critical two-tail	2.1604	

the average score of G1 is 5.34, indicating that G2 performed better than G1 in the questionnaire test stage. PTVis is better than the text summary in helping users acquire knowledge about the penetration testing process. Compared with the text summary, PTVis allows users to explore the different stages of penetration testing by simulating penetration testing scenarios, which is more conducive to a multi-angle and in-depth analysis of the characteristics of penetration testing.

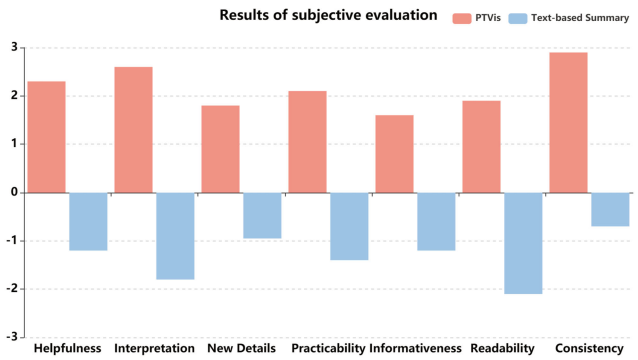


FIGURE 12. Subjective evaluation focuses on users' psychological acceptance of using PTVis and the text-based summary. Each participant completed the questionnaire by answering on a 7-point Likert scale.

In the subjective evaluation, G1 and G2 exchanged experimental content to compare the text-based summary and PTVis. A 7-point Likert scale was used to evaluate the two approaches, and to calculate the deviation value of the evaluation results, as shown in Figure 12. Most participants believed that PTVis has advantages in terms of helpfulness, interpretation, new details, practicality, informativeness, and consistency. However, in terms of readability, the absolute score for PTVis is 1.9, while the absolute score of the text-based summary score is 2.1. The reason is that visualization requires participants to comprehend for a certain amount of time. To evaluate the extent to which PTVis plays a role and to evaluate which visual elements are considered critical, we developed two related questions:

RQ1: How does PTVis help participants comprehend the penetration testing process?

RQ2: What refinement measures are applied to the automatically generated results?

In addition, the observation results and user feedback with regard to RQ1 and RQ2 are discussed in the following (*G* represents the group, and *P* represents the number within the group).

1) RQ1: HOW DOES PTVis HELP PARTICIPANTS COMPREHEND THE PENETRATION TESTING PROCESS?

a: CONSTRUCT A PENETRATION TESTING PATH TO INCREASE AWARENESS

All users believed that the construction of the penetration testing path allowed them to newly comprehend penetration testing to a great extent and to notice many details that were not thought of before. This was also the visual feature that was first noticed. "*When we look at this process from a global perspective, we may say, 'Oh, the original penetration testing process is from information gathering to attack execution, and the data information is used as a correlation between the steps'*" (G1P4). "*It is clear which penetration testing steps are related to each other and the order in which the steps are executed. Therefore, the construction of the path really helps to explore penetration testing methods*" (G2P3). "*The way that I view the results of a single step while choosing the penetration testing steps and tools gives me a sense of participation*" (G2P13).

Most users could quickly comprehend the flow of the penetration testing method, the composition of the steps and the use of tools by constructing a penetration testing path. After a penetration testing path is constructed, most penetration testing methods can be remembered. Five users suggested that PTVis should be more detailed in the step division and that constructing a penetration testing path with more detailed steps would be helpful for deeper comprehension.

b: THE EVALUATION MODEL BRINGS MORE DETAILS

Regarding the evaluation of penetration testing tools, the feedback was uniformly positive, and the users believed that the evaluation results promote the construction of penetration testing paths. "*The evaluation of tools is intuitive, and you can easily analyse the differences between different tools*" (G1P8). "*I think the tool evaluation is really useful and can help me choose a tool to a great extent. Whether I consider it as a whole or in terms of one or more characteristics, I can quickly choose the right tool*" (G2P7).

All users reviewed the evaluation results of the penetration testing tools. One user suggested providing more diversified penetration testing tools for selection and that the tools between different steps should be as different as possible to help users understand as many related tools as possible.

c: BACKTRACKING THE RESULTS TO DEEPEN COMPREHENSION

Penetration testing results organize events in the order of penetration testing path construction. Users can backtrack the construction process of the penetration testing paths and perform multiple analyses. "*Thanks to the result backtracking, I am deeply impressed with the specific scenarios or specific results of the constructed penetration testing path, and I can remember my thoughts when building the path*" (G1P14). "*Although I have a good comprehension of the exploration of penetration testing methods, backtracking is definitely helpful. Multiple backtracking analysis helps me find some subtle details I overlooked*" (G1P6). "*I think it is helpful to backtrack the construction process through the results, especially the step time. It makes me realize that I usually ignore efficiency when I choose a tool*"(G2P10). More than half of the users utilized the penetration testing results interface to backtrack the construction process of the penetration testing paths. They believe that the connection between the process and the results of penetration testing is more integrated based on retrospective analysis and that the construction of the penetration testing paths can be optimized.

The feedback of most users regarding the classification of the threat level of the steps, which can help to identify the steps with high threat levels, was positive. Additionally, some users suggested that the threat level of the steps should be reflected in the path construction. "*I think that the path construction should also show the threat level of the penetration testing steps, not just distinguish between 'information gathering execution' and 'penetration attack execution' so that we can better understand the threat levels of different penetration testing methods*" (G2P2).

2) RQ2: WHAT REFINEMENT MEASURES ARE APPLIED TO THE AUTOMATICALLY GENERATED RESULTS?

All users performed operations to judge the success of the penetration testing path and to add a path to the penetration testing result interface. Nine users deleted the nodes they thought were unnecessary in the construction of the penetration testing path. "*I deleted these nodes because I found that a different penetration testing method can get results faster*" (G1P10). Only a few users (5/28) selectively added penetration testing paths to the result interface. They thought that the construction process is relatively successful, and they fully comprehended the penetration testing method. In their view, these methods do not need backtracking. Thus, these users were more inclined to analyse unfamiliar methods. Most users believed that the visual design and interactive operation of PTVis make the penetration testing process easier to comprehend. One user commented that the design of PTVis makes it possible to effectively summarize the penetration testing process: "*If I want to conduct stakeholder communication about penetration testing in my organization, it will be really helpful to show it with PTVis*" (G2P5).

D. DISCUSSION

We discuss the design inspiration in user research and propose reasonable future research work.

1) NARRATIVE STRUCTURE IN PENETRATION TESTING SUMMARY

PTVis directly maps six selected penetration testing interpretive elements to visual encoding. Some of the subtle characteristics shown by these elements are useful for analysing penetration testing data. At the same time, the participants developed a question: the PTVis is limited in showing a more detailed context and data generated by the step nodes.

To address this problem, the architecture of different penetration testing modes can be explored to allow more diversity and complexity in future work. In PTVis, based on different prerequisites and penetration testing scenarios, a variety of penetration testing elements can be selected with the corresponding architecture to further organize and construct interpretive mapping and to characterize different penetration testing modes in more detail from multiple perspectives.

2) VISUAL ENCODING OF PENETRATION TESTING ELEMENTS

In the design of PTVis, the process decision tree diagram and penetration testing results Sankey diagram characterize the penetration testing paths, and the word cloud characterizes the extracted keywords. Based on user feedback, these visual encoding can be improved.

In future work on PTVis, the storyline visualization technique [55] can be used to characterize the penetration testing process. One of the advantages of this technique is that it can branch and merge lines along the vertical axis as events advance. PTVis can use ''Step-Event'' as the theme to construct a penetration testing path, integrate the information gathering methods of different penetration testing methods, and realize different penetration testing paths from the same information gathering step node to construct in parallel. Visualization based on the ''Step-Event'' theme has fewer restrictions. Users can customize the construction of penetration testing paths based on the information gathering results. The merging, branching, and crossing over of paths can indicate the connection between different penetration testing methods and the changes in user penetration testing thinking, promoting a more detailed exploration of the penetration testing process. With regard to the word cloud, other visual channels can be explored, such as font colour saturation, the use of extra tags, and spatial layout [47], to show the importance of keywords.

VII. CONCLUSION

Analysing the penetration testing process can promote the development of network security technology, and penetration testing can truly simulate network intrusion actions, serving as a beneficial supplement to network security assessment. Therefore, it is advisable to promote a deeper comprehension and analysis of the penetration testing process, for example, through PTVis. This paper presents an interpretive approach to help users analyse and comprehend the penetration testing process. We use a visual, narrative approach to correlate and present the primary interpretive elements of the penetration testing process in a lightweight, intuitive, and informational form and to facilitate customized exploration of penetration testing methods through the auxiliary decision to further improve the interpretability and comprehensibility of penetration testing. Finally, the experimental analysis proves the feasibility and effectiveness of our work.

ACKNOWLEDGMENT

The authors would like to thank the University of Electronic Science and Technology of China (UESTC) and i-Chunqiu Company to support the dataset and penetration testing technique.

REFERENCES

[1] S. Hariri, T. Dharmagadda, M. Ramkishore, G. Qu, and C. S. Raghavendra, ''Vulnerability analysis of faults/attacks in network centric systems,'' in *Proc. ISCA PDCS*, 2003, pp. 256–261.
[2] J. Yeo, ''Using penetration testing to enhance your company's security,'' *Comput. Fraud Secur.*, vol. 2013, no. 4, pp. 17–20, Apr. 2013.
[3] J. P. Mcdermott, ''Attack net penetration testing,'' in *Proc. Workshop New Secur. Paradigms*, 2001, pp. 15–21.
[4] Y. Shi, C. Bryan, S. Bhamidipati, Y. Zhao, Y. Zhang, and K.-L. Ma, ''MeetingVis: Visual narratives to assist in recalling meeting context and content,'' *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 6, pp. 1918–1929, Jun. 2018, doi: 10.1109/TVCG.2018.2816203.
[5] R. D. Matz, J. Rohwer, and D. William, ''Visual elaboration and comprehension of text,'' *Audiovisual Aids*, vol. 14, pp. 14–29, Jan. 1971.
[6] G. Conti and K. Abdullah, ''Passive visual fingerprinting of network attack tools,'' in *Proc. ACM Workshop Vis. Data Mining Comput. Secur.*, 2004, pp. 45–54, doi: 10.1145/1029208.1029216.
[7] A. Singh, L. Bradel, A. Endert, R. Kincaid, C. Andrews, and C. North, ''Supporting the cyber analytic process using visual history on large displays,'' in *Proc. 8th Int. Symp. Vis. Cyber Secur. VizSec*, 2011, pp. 1–8, doi: 10.1145/2016904.2016907.
[8] M. Angelini, G. Blasilli, P. Borrello, E. Coppa, D. C. DrElia, S. Ferracci, S. Lenti, and G. Santucci, ''ROPMate: Visually assisting the creation of ROP-based exploits,'' in *Proc. IEEE Symp. Vis. Cyber Secur. (VizSec)*, Oct. 2018, pp. 1–8, doi: 10.1109/VIZSEC.2018.8709204.
[9] J. Yuen, B. Turnbull, and J. Hernandez, ''Visual analytics for cyber red teaming,'' in *Proc. IEEE Symp. Vis. Cyber Secur. (VizSec)*, Oct. 2015, pp. 1–8, doi: 10.1109/VIZSEC.2015.7312765.
[10] P. Herzog, *The Open Source Security Testing Methodology Manual*. New York, NY, USA: ISECOM, 2015. [Online]. Available: http://www.isecom.org/
[11] Open Information Systems Security Group. (2006). *Information Systems Security Assessment Framework*. [Online]. Available: http://www.oissg.org/
[12] The Ptes Team. (2017). *The Penetration Testing Execution Standard Documentation*. [Online]. Available: http://www.pentest-standard.org/
[13] H. Shiravi, A. Shiravi, and A. A. Ghorbani, ''A survey of visualization systems for network security,'' *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 8, pp. 1313–1329, Aug. 2012, doi: 10.1109/TVCG.2011.144.
[14] Z. Ying, F. Xiaoping, F. Zhou, and J. Zhang, ''A survey on network security data visualization,'' *J. Comput.-Aided Des. Comput. Graph.*, vol. 26, no. 5, pp. 687–697, 2014.
[15] Y. Shi, Y. Zhao, F. Zhou, R. Shi, Y. Zhang, and G. Wang, ''A novel radial visualization of intrusion detection alerts,'' *IEEE Comput. Graph. Appl.*, vol. 38, no. 6, pp. 83–95, Nov. 2018, doi: 10.1109/MCG.2018.2879067.
[16] M. Angelini, G. Blasilli, T. Catarci, S. Lenti, and G. Santucci, ''Vulnus: Visual vulnerability analysis for network security,'' *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 1, pp. 183–192, Jan. 2019, doi: 10.1109/TVCG.2018.2865028.
[17] R. Ball, G. A. Fink, and C. North, ''Home-centric visualization of network traffic for security administration,'' in *Proc. ACM Workshop Vis. Data Mining Comput. Secur. (VizSEC/DMSEC)*, 2004, pp. 55–64, doi: 10.1145/1029208.1029217.
[18] F. Mansmann, D. Keim, S. North, B. Rexroad, and D. Sheleheda, ''Visual analysis of network traffic for resource planning, interactive monitoring, and interpretation of security threats,'' *IEEE Trans. Vis. Comput. Graphics*, vol. 13, no. 6, pp. 1105–1112, Nov. 2007, doi: 10.1109/TVCG.2007.70522.
[19] J. McPherson, K.-L. Ma, P. Krystosk, T. Bartoletti, and M. Christensen, ''PortVis: A tool for port-based detection of security events,'' in *Proc. ACM Workshop Vis. Data Mining Comput. Secur. (VizSEC/DMSEC)*, 2004, pp. 73–81, doi: 10.1145/1029208.1029220.

[20] D. Plonka, "FlowScan: A network traffic flow reporting and visualization tool," in *Proc. LISA*, Jan. 2000, pp. 305–317.

[21] K. Lakkaraju, W. Yurcik, and A. J. Lee, "NVisionIP: Netflow visualizations of system state for security situational awareness," in *Proc. ACM Workshop Vis. Data Mining Comput. Secur. (VizSEC/DMSEC)*, 2004, pp. 65–72, doi: 10.1145/1029208.1029219.

[22] Y. Zhao, F. Zhou, X. Fan, X. Liang, and Y. Liu, "IDSRadar: A real-time visualization framework for IDS alerts," *Sci. China Inf. Sci.*, vol. 56, no. 8, pp. 1–12, Aug. 2013, doi: 10.1007/s11432-013-4891-9.

[23] H. Shiravi, A. Shiravi, and A. Ghorbani, "Situational assessment of intrusion alerts: A multi attack scenario evaluation," in *Proc. Int. Conf. Inf. Commun. Secur.*, Nov. 2011, pp. 399–413, doi: 10.1007/978-3-642-25243-3_32.

[24] S. McKenna, D. Staheli, C. Fulcher, and M. Meyer, "BubbleNet: A cyber security dashboard for visualizing patterns," *Comput. Graph. Forum*, vol. 35, no. 3, pp. 281–290, Jun. 2016, doi: 10.1111/cgf.12904.

[25] S. Walton, E. Maguire, and M. Chen, "A visual analytics loop for supporting model development," in *Proc. Symp. Vis. Cyber Secur.*, Oct. 2015, pp. 1–8, doi: 10.1109/VIZSEC.2015.7312767.

[26] L. Briesemeister, S. Cheung, U. Lindqvist, and A. Valdes, "Detection, correlation, and visualization of attacks against critical infrastructure systems," in *Proc. 8th Int. Conf. Privacy, Secur. Trust*, Sep. 2010, pp. 15–22, doi: 10.1109/PST.2010.5593242.

[27] B. Arkin, S. Stender, and G. McGraw, "Software penetration testing," *IEEE Secur. Privacy Mag.*, vol. 3, no. 1, pp. 84–87, Jan. 2005, doi: 10.1109/MSP.2005.23.

[28] *2011 Systems and Software Engineering–Systems and Software Quality Requirements and Evaluation (Square)–System and Software Quality Models*, Standard ISO/IEC 25010, 2013.

[29] P. Lew, L. Olsina, and L. Zhang, "Quality, quality in use, actual usability and user experience as key drivers for Web application evaluation," in *Proc. Int. Conf. Web Eng.*, Jul. 2010, pp. 218–232, doi: 10.1007/978-3-642-13911-6_15.

[30] M. Herrera, M. Moraga, I. Caballero, and C. Calero, "Quality in use model for Web portals (QiUWeP)," in *Proc. Int. Conf. Web Eng.*, Jul. 2010, pp. 91–101, doi: 10.1007/978-3-642-16985-4_9.

[31] T. Marir, F. Mokhati, and H. Bouchlaghem-Seridi, "Do we need specific quality models for multi-agent systems?—Toward using the ISO/IEC 25010 quality model for MAS," in *Proc. 9th Int. Conf. Softw. Eng. Appl.*, 2014, pp. 363–368.

[32] L. Bautista, A. Abran, and A. April, "Design of a performance measurement framework for cloud computing," *J. Softw. Eng. Appl.*, vol. 05, no. 02, pp. 69–75, 2012, doi: 10.4236/jsea.2012.52011.

[33] B. Zeiss, D. Vega, I. Schieferdecker, H. Neukirchen, and J. Grabowski, "Applying the ISO 9126 quality model to test specifications–exemplified for TTCN-3 test specifications," in *Proc. Softw. Eng., Fachtagung des GI-Fachbereichs Softwaretechnik*. Hamburg, Germany: GI, Mar. 2007, pp. 231–244. [Online]. Available: https://dl.gi.de/20.500.12116/22765

[34] B. Behkamal, M. Kahani, and M. K. Akbari, "Customizing ISO 9126 quality model for evaluation of B2B applications," *Inf. Softw. Technol.*, vol. 51, no. 3, pp. 599–609, Mar. 2009, doi: 10.1016/j.infsof.2008.08.001.

[35] Y. Zongkui, Z. Jinhui, X. Liu, and Q. Wang, "Software quality evaluation system based on ISO/IEC 9126 standard," *J. Xidian Univ.*, vol. 31, no. 1, pp. 47–53, 2004.

[36] OWASP. (2017). *Open Web Application Security Project Top 10*. [Online]. Available: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

[37] K. Cook and J. Thomas, *Illuminating the Path: The Research and Development Agenda for Visual Analytics*, vol. 54. New York, NY, USA: IEEE, May 2005.

[38] E. Hutchins, M. Cloppert, and R. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Leading Issues in Information Warfare and Security Research*, vol. 1. New York, NY, USA: Academic, Jan. 2011.

[39] D. Broomfield and G. M. Humphris, "Using the delphi technique to identify the cancer education requirements of general practitioners," *Med. Educ.*, vol. 35, no. 10, pp. 928–937, Oct. 2001.

[40] H. C. Choi and E. Sirakaya, "Sustainability indicators for managing community tourism," *Tourism Manage.*, vol. 27, no. 6, pp. 1274–1289, Dec. 2006.

[41] M. Joseph and L. Aamir, *Web Penetration Testing With Kali Linux*, no. 2. Birmingham, U.K.: Packt, 2019, doi: 10.4324/9781351309240-4.

[42] W. Kintsch, *Comprehension: A Paradigm for Cognition*. Cambridge, U.K.: Cambridge Univ. Press, Jan. 1998.

[43] D. A. Keim, G. Andrienko, J. Fekete, C. Gorg, J. Kohlhammer, and G. Melancon, "Visual analytics: Definition, process, and challenges," in *Information Visualization* (Lecture Notes in Computer Science), vol. 4950. Berlin, Germany: Springer, 2008, pp. 154–175, doi: 10.1007/978-3-540-70956-5_7.

[44] D. Sinha. (2019). Dark Mode—What is It, and Why Do We Need It? Techahead. [Online]. Available: https://www.techaheadcorp.com/blog/dark-mode/

[45] K. Ferris and S. Zhang, "A framework for selecting and optimizing color scheme in Web design," in *Proc. 49th Hawaii Int. Conf. Syst. Sci. (HICSS)*, Jan. 2016, pp. 532–541.

[46] C. Wei, Z. Shen, and Y. Tao, *Data Visualization*. Beijing, China: Publishing House of Electronics Industry, 2013.

[47] C. Felix, S. Franconeri, and E. Bertini, "Taking word clouds apart: An empirical investigation of the design space for keyword summaries," *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 1, pp. 657–666, Jan. 2018, doi: 10.1109/TVCG.2017.2746018.

[48] S. Lohmann, F. Heimerl, F. Bopp, M. Burch, and T. Ertl, "Concentri cloud: Word cloud visualization for multiple text documents," in *Proc. 19th Int. Conf. Inf. Vis.*, Jul. 2015, pp. 114–120.

[49] F. Heimerl, S. Lohmann, S. Lange, and T. Ertl, "Word cloud explorer: Text analytics based on word clouds," in *Proc. 47th Hawaii Int. Conf. Syst. Sci.*, Jan. 2014, pp. 1833–1842.

[50] T. Munzner, *Visualization Analysis and Design*. Boca Raton, FL, USA: CRC Press, 2014.

[51] C. Liu and P. Wang, "A sunburst-based hierarchical information visualization method and its application in public opinion analysis," in *Proc. 8th Int. Conf. Biomed. Eng. Informat. (BMEI)*, Oct. 2015, pp. 832–836, doi: 10.1109/BMEI.2015.7401618.

[52] T. Merčun, M. Žumer, and T. Aalberg, "Presenting bibliographic families using information visualization: Evaluation of FRBR-based prototype and hierarchical visualizations," *J. Assoc. Inf. Sci. Technol.*, vol. 68, no. 2, pp. 392–411, Feb. 2017, doi: 10.1002/asi.23659.

[53] M. Bruls, K. Huizing, and J. J. V. Wijk, *Squarified Treemaps*. Vienna, Austria: Springer, 2000.

[54] M. Luchetti and A. R. Sutin, "Measuring the phenomenology of autobiographical memory: A short form of the memory experiences questionnaire," *Memory*, vol. 24, no. 5, pp. 592–602, May 2016.

[55] Y. Tanahashi and K.-L. Ma, "Design considerations for optimizing storyline visualizations," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 12, pp. 2679–2688, Dec. 2012, doi: 10.1109/TVCG.2012.212.

SIJIE ZHENG received the B.S. degree from the Southwest University of Science and Technology, Mianyang, China, in 2018, where he is currently pursuing the M.S. degree with the School of Computer Science. His research interests include network security visualization and progressive visual analysis.

YADONG WU (Member, IEEE) received the B.S. degree from Zhengzhou University, in 1999, the M.S. degree from the Southwest University of Science and Technology, in 2003, and the Ph.D. degree from the University of Electronic Science and Technology of China, in 2008. He was a Visiting Scholar with the LE2I, University of Burgundy, France, in 2009, and the University of Central Arkansas, USA, in 2011. He is currently a Professor with the Sichuan University of Science and Engineering and the Southwest University of Science and Technology (ESUST). He is also the Dean of the School of Computer Science and Engineering, ESUST. He is also a Senior Member of the Chinese Society of Image Graphics (CSIG), and the Deputy Secretary of CSIG Visualization and Visual Analysis Special Committee. He was has been the Chairman of ChinaVis since 2019, CCF YOCSEF Chengdu from 2016 to 2017, and CCF Mianyang since 2016. He has published more than 200 articles in domestic and international journals, such as *Electronic Journal*, *Neurocomputing*, *Journal of Visualization*, and so on. His current research interests include visualization and visual analysis, human–computer interaction, and digital image processing.

SONG WANG received the Ph.D. degree from the Institute of Electronic Engineering, China Academy of Engineering Physics, in 2019. He is currently the Deputy Director of the Department of Software Engineering, School of Computer Science and Technology, Southwest University of Science and Technology. He presided over one National Natural Science Youth Fund Project, one Sichuan Miaozi Project, and one Doctoral Fund Project. In the past three years, he has published more than ten academic articles in foreign journals and conferences, six of which were indexed by SCI/EI, applied for six invention patents, and obtained five software copyrights. His research interests include visualization and visual analysis, research content includes flow field visualization, medical visualization, network security visualization, urban computing visualization, and so on. He is also a member of IEEE CS, CCF, and CSIG.

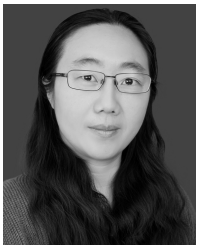
YONG WEI received the Ph.D. degree from the University of Science and Technology of China, in 2009. He is currently a Teacher with the School of Computer Science and Technology, Southwest University of Science and Technology. He presided over or participated in one national 863 projects, 1, 242 projects, one major special sub-project of the Ministry of Science and Technology, one national natural science youth fund project, one project of the Sichuan Provincial Department of Education, and more than ten horizontal projects. He has participated in the compilation of one academic monograph, published four academic articles, applied for one invention patent, and applied for one national defense patent. His research interests include network security situation awareness, network security content monitoring, network security visualization, and so on. He is also a member of CCF.

DONGSHENG MU received the B.S. degree from China West Normal University, Nanchong, China, in 2018. He is currently pursuing the M.S. degree with the School of Computer Science, Southwest University of Science and Technology, Mianyang, China. His research interest includes information visualization.

HUAN HE is currently pursuing the B.S. degree with the School of Computer Science, Southwest University of Science and Technology, Mianyang, China. His research interest includes information visualization.

DONGXUAN HAN received the B.S. degree in automation from the Southwest University of Science and Technology, Mianyang, China, in 2017, where he is currently pursuing the M.S. degree with the School of Computer Science. He was a Visiting Student with the Illinois Institute of Technology, Chicago, USA, in 2016. His research interests include urban data visual analysis, especially trajectory visual analysis.

JING LIAO received the M.A. degree from the Southwest University of Science and Technology (SWUST), in 2011. He is currently a Lecturer with SWUST. His research interests include data visualization and visual analysis.



HUARONG CHEN received the M.A. degree from UESTC, in 2008. She is currently a Lecturer with the Southwest University of Science and Technology (SWUST). She presided over one project of the Sichuan Provincial Department of Science and Technology, led and participated in more than 40 projects at various levels, including the National Natural Science Foundation of China and national defense research, and published more than ten articles in journals, such as the *Journal of University of Electronic Science and Technology of China*, and obtained software copyright three items, one item of provincial and municipal scientific research awards, and four items of provincial school teaching achievement awards. Her research interests include data visualization and visual analysis.