Desieve the Attacker: Thwarting IP Theft in Sieve-Valve-based Biochips

Mohammed Shayan*, Sukanta Bhattacharjee†, Yong-Ak Song*†, Krishnendu Chakrabarty‡, and Ramesh Karri*
*New York University, †New York University Abu Dhabi, ‡Duke University
Email: mos283@nyu.edu, sb6538@nyu.edu, rafael.song@nyu.edu, krish@ee.duke.edu, rkarri@nyu.edu

***Abstract*—Researchers develop bioassays following rigorous experimentation in the lab that involves considerable fiscal and highly-skilled-person-hour investment. Previous work shows that a bioassay implementation can be reverse engineered by using images or video and control signals of the biochip. Hence, techniques must be devised to protect the intellectual property (IP) rights of the bioassay developer. This study is the first step in this direction and it makes the following contributions: (1) it introduces use of a sieve-valve as a security primitive to obfuscate bioassay implementations; (2) it shows how sieve-valves can be used to obscure biochip building blocks such as multiplexers and mixers; (3) it presents design rules and security metrics to design and measure obfuscated biochips. We assess the cost-security trade-offs associated with this solution and demonstrate practical sieve-valve based obfuscation on real-life biochips.**

I. Introduction

A biochip platform integrates complex laboratory operations into a small chip of few square centimeters in size. It has revolutionized biochemical applications such as point-of-care diagnostics [1], DNA purification [2], and biomedical research [3]. The microfluidics market is valued at $8.28 Billion in 2017 and it is expected to grow at a compound annual growth rate of 22.6% to reach $27.91 Billion by 2023 [4]. Due to rapid commercialization and deployment, intellectual property (IP) piracy has become financially rewarding [5]. Therefore, protecting bioassay IPs is of paramount importance to its developers.

Pharmaceutical companies invest large sums of money and man-hours in a slow and expensive drug development process laced with tough regulations. This process is prone to stealing of sensitive research data [6]. In 2016, two scientists at a leading pharmaceutical company were indicted for colluding with a competitor to steal promising drug research secrets [7]. For rapid and low-cost drug development, pharmaceutical companies are using various types of microfluidic biochips that minimize the assay time and reagent requirement [8].

Continuous flow-based microfluidic biochips (CFMBs) have evolved rapidly in the last decades [3], [9]. The CFMBs allow automated control of fluid flow in a network of micro-channels by suitable actuation of pressure driven micro-valves [9]. Previous work has shown that a bioassay implementation on a CFMB can be reverse engineered using biochip images and actuation sequence [10]. Therefore, there is a need to devise methods that protect the bioassay IP implementation on biochips, which in turn is critical for the successful adaptation of biochips. This work is a first step towards protecting bioassay IP based on hardware primitives on a CFMB. Our contributions are summarized as follows:

- We propose sieve-valve based obfuscation of a bioassay to defeat bioassay reverse engineering (RE) [10].
- We show how sieve-valves can obfuscate biochip building blocks such as channels, multiplexers, and mixers.
- We develop design rules, security metrics, and cost trade-offs for a sieve-valve-based obfuscated biochip.
- We demonstrate the practicality of the obfuscation method by applying it to a real-life biochip benchmark.

The rest of the paper is organized as follows. Section II provides a motivational example to describe the IP piracy threat. Section III provides the relevant background and Section IV describes the use of sieve-valves to achieve bioassay obfuscation. Section V develops the metrics and design rules associated with the sieve-valve based obfuscation. Section VI provides experimental results of obfuscation applied to real-life biochips and Section VII concludes the paper.

II. Motivation

We demonstrate IP piracy through a bioassay implementation on a CFMB (Fig. 1). The platform consists of a multiplexer that selects from two input reagents R_1 and R_2 and uses a rotary mixer to mix them in the desired ratio [11]. Fluidic operations corresponding to a bioassay are mapped to a sequence of actuation steps for controlling the valve state. Let us illustrate the bioassay execution on the CFMB in which all valves are initially closed. The first set of actuations fills R_1 in the upper half of the mixer (Fig. 1(a)). Next, R_2 fills the lower half of the mixer is filled with (Fig. 1(b)). The valves $6, 7, 8$ are activated in a sequence to form a peristaltic pump that circulates the fluid in the rotary mixer, producing a mixture of R_1 and R_2 in 1:1 ratio (Fig. 1(c)). Next, the lower half of the mixer is replaced with R_1 (Fig. 1(d)), and the peristaltic pump is activated (Fig. 1(e)). The resulting fluid contains R_1 and R_2 in 3:1 ratio (Fig. 1(f)).

Fig. 1 shows the straightforward one-to-one mapping between the actuation sequence, biochip snapshots, and fluidic operations. It can be inferred from the biochip snapshots that the bioassay mixes two input fluids in a 3:1 ratio [10]. The corresponding sequencing graph (IP) is shown in Fig. 1(e). The mixing time can also be determined from the actuations. This example demonstrates the ease with which the bioassay description and its parameters can be reverse engineered [10]. To thwart the RE of bioassays, we need to obfuscate the one-to-one mapping between the actuation sequence, biochip snapshots, and fluidic operations.

III. Background

In this section, we present the background on CMFBs and recent work on bioassay IP protection.

A. Continuous-Flow-Microfluidic Biochips

CFMB consists of two layers of permanently etched micro-channels called the flow and the control layer, as shown in Fig. 2(a). At the intersection of the two layers, a "valve" is formed. An external pressure source can control this valve.

978-3-9819263-2-3/DATE19/©2019 EDAA

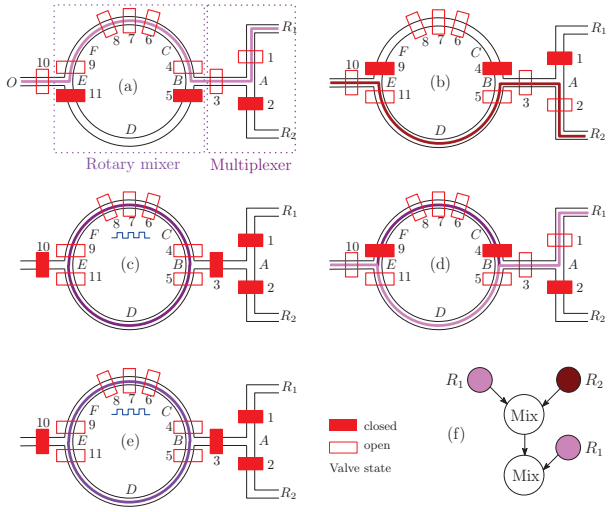

Fig. 1: A bioassay implementation: (a) Push reagent R_1 into the upper half of mixer. (b) Push reagent R_2 into the lower half of mixer. (c) Mix. (d) Push R_1 into the lower half of mixer. (e) Mix. (f) Sequencing graph inferred from the images.

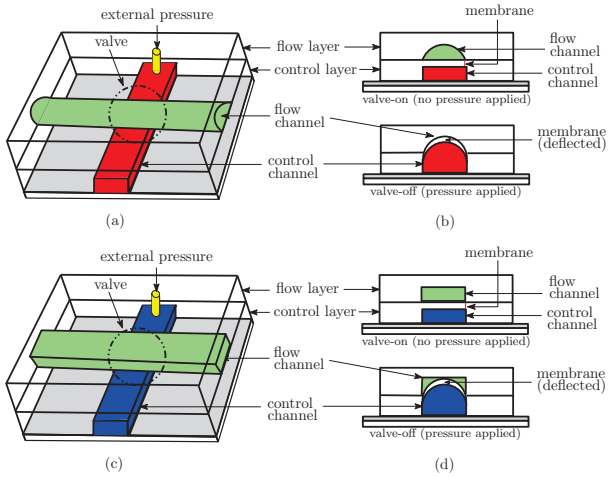

Fig. 2: Schematic of a two-layer microfluidic device: ordinary valve (a) top view and (b) cross-section view; Sieve valve: (c) top view and (d) cross-section view.

When the valve is pressurized, the flexible membrane of the control layer deflects deep into the flow layer blocking the fluid flow (Fig. 2(b)). By opening/closing of the valves, complex fluid handling operations such as mixing, incubation, transportation, and storage can be performed [9]. Advancement in multi-layer soft lithography techniques enables thousands of valves to be integrated into a tiny chip [9].

In a normal valve, the flow channel is semi-circular shaped. When the valve is pressurized, it seals the flow channel (Fig. 2(b)). However, if the flow channel is rectangular, the pressurized valve membrane partially closes the flow channel, as shown in Fig. 2(d). This is the sieve valve (Fig. 2(c)) [12]. These are used in CFMBs to trap cells. Closing the sieve valve blocks the cells but allows the fluid to pass through [13].

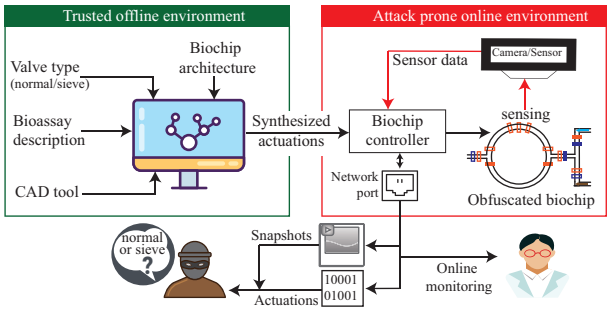

Fig. 3: Proposed sieve-valve-based obfuscation technique.

B. Related Work

An assessment of IP threats in the supply chain due to the distributed microfluidic design flow is presented in [5]. The threats include overbuilding, RE, and counterfeiting of biochips. A bioassay locking scheme was proposed to obfuscate the sequencing graph description of the bioassay [14]. However, this technique does not prevent or resist RE of a bioassay from the corresponding actuation sequence. A method for camouflaging the biochip layout by inserting extra valves and channels is reported in [10]. However, this approach fails as an attacker can RE the IP by combining the actuation sequence and biochip layout.

IV. Obfuscation for IP Protection

In this section, we present the threat model, sieve-valve-based obfuscation methodology and two types of obfuscation - behavioral and structural.

A. Threat Model

Where? — Consider a bioassay developer who invests heavily in bioassay IP development and uses microfluidic platforms to conduct large-scale experiments involving the bioassay [3].
Who? — A competitor is motivated to steal the IP from the developer without incurring any cost of development.
What? — To RE the bioassay, the attacker accesses the actuation sequence and the video or snapshots of the biochip captured by the camera sensor.
How? — An attacker can get the "what" through a network attack. The biochip controller is connected to the network for round-the-clock online monitoring and control [15]. A remote attacker can gain administrator credentials using a social engineering attack or malware [16], [17]. Alternately, an attacker can collude with rogue insiders to capture the video of bioassay execution. Image analysis based RE can recover the actuation sequence and the bioassay [10].
Limitations — The attacker can differentiate between actuated valve and deactivated valve due to the visual difference. However, the attacker does not know which one is a normal valve or which one is sieve valve as the top view of the both is identical (Fig. 2). Further, the attacker cannot obverse the flow of fluid through the channels as it does not produce any visual difference. The attacker does not have access to the CFMB.

B. Sieve-valve-based Obfuscation

To deter the RE of a bioassay, we propose to obfuscate the actuation sequence by carefully inserting sieve valves in the biochip. The bioassay developer keeps the bioassay description

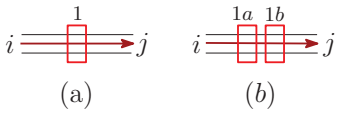

Fig. 4: Channel $i \rightarrow j$ with (a) valve 1 and (b) valves $1a$, $1b$.

and the sieve-valve locations a secret. The developer uses a CAD tool on a trusted offline computer to synthesize the obfuscated actuation sequence. The obfuscated sequence is loaded in the biochip controllers that are used to conduct the high-valued-experiments, as shown in Fig. 3. Minor software updates are handled in the biochip controller and major updates are performed in the trusted offline computer.

TABLE I: Boolean variables to characterize a channel.

Parameter	Description	Interpretation	
		1	0
v_a	Status of a valve	actuated	unactuated
g_a	Type of a valve	normal	sieve
$c_{ij}^k, k \in \mathbb{N}$	Status of channel $i \rightarrow j$	open	closed

k: #valves on the fluidic channel connecting ports i and j.

Consider the channel between port i and j, as shown in Fig. 4(a). Let the channel $i \rightarrow j$ be open if the valve 1 is actuated, else it is closed. Such a valve is a normal valve. On the other hand, if the valve is a sieve, then the channel $i \rightarrow j$ is always open, regardless of the actuation state of valve 1. To capture the differences between a normal and a sieve valve, consider the Boolean variables defined in Table I. Using these variables, we describe the channel in Fig. 4(a) as:

$$c_{ij}^1 = \overline{g_1} + g_1 \cdot v_1 = \overline{g_1} + v_1 \tag{1}$$

As per the attack model, g_1 is secret, v_1 is known from the actuation sequence. Equation (1) captures the obfuscation introduced in the fluid channel characteristics due to the unknown valve type. Without the knowledge of g_1, an attacker does not know the channel status. The sieve valve reduces the flow-rate in the channel. However, this can be neutralized by either increasing the pressure or allowing more time for the fluid flow. For the purposes of our analysis, we ignore the change in flow-rate.

Consider an increase in the number of valves on the channel, as shown in Fig. 4(b). The characteristic of the channel is given by following Boolean equation,

$$\begin{aligned} c_{ij}^2 &= (\overline{g_{1a}} + v_{1a}) \cdot (\overline{g_{1b}} + v_{1b}) \\ &= \overline{g_{1a}} \cdot \overline{g_{1b}} + \overline{g_{1a}} \cdot v_{1b} + \overline{g_{1b}} \cdot v_{1a} + v_{1a} \cdot v_{1b} \end{aligned} \tag{2}$$

If there are n such valves on a channel $i \rightarrow j$, the characteristic of the channel can be captured as

$$c_{ij}^n = \bigwedge_{\gamma=1}^{n} \left(\overline{g_\gamma} + v_\gamma\right) \tag{3}$$

Comparing Equation (2) and Equation (3), increasing the number of valves increases the channel obfuscation due to the corresponding increase in the number of unknown parameters (g_*). Using this primitive, we describe two types of biochip obfuscation: behavioral and structural.

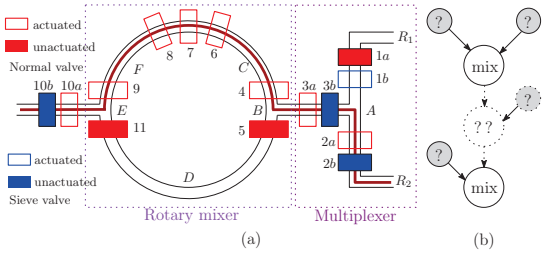

Fig. 5: (a) Sieve-valve-based obfuscated functional modules of mixer and multiplexer. (b) The obfuscated sequencing graph.

C. Behavioral Obfuscation

A biochip consists of functional modules such as a fluid inlet/outlet, mixer, storage, reaction chamber, and multiplexer/demultiplexer. As shown in Section II, the actuation signals of a biochip have a one-to-one mapping to the fluidic operations. We insert sieve valves in the biochip functional modules so that the actuation-signal to fluidic-operation mapping is no longer preserved. Since the valve type is kept secret, the channel characteristic can be obfuscated, as shown in Equation (3). Thus, the attacker cannot determine the fluidic operations correctly to RE the sequencing graph (IP). This is called *behavioral obfuscation*.

Consider the biochip shown in Fig. 1 with a two-input multiplexer and a rotary mixer. It mixes two input reagents R_1 and R_2 in a $3:1$ ratio, as explained in Section II. Additional valves (normal and sieve) are added to obfuscate the biochip, as shown in Fig. 5(a). In the modified CFMB platform, one or more sieve valves on the input-to-output paths can be unactuated to deceive the attacker from identifying the correct fluidic path. From the Equation (3), the channel state (open/close) depends on the valve type (g_*), which is unknown to the attacker. The following example illustrates obfuscation on the fluidic path.

Example 1. *In Fig. 5(a), let* $\{1b, 2b, 3b, 10b\}$ *be sieve valves and the rest be normal valves, i.e.,* $g_{1b}, g_{2b}, g_{3b}, g_{10b} = 0000$*, then the actuation set* $v_{1a}, v_{1b}, v_{2a}, \cdots, v_{10a}, v_{10b}, v_{11} = 011010101111110$ *pushes* R_2 *into the mixer (ref. Fig. 5(a)). On the other hand, if* $\{1a, 2a, 3b, 10b\}$ *are sieve valves and rests are normal valves, then the same actuation set will push* R_1 *into the mixer. Without knowing the valve type (sieve or normal), an attacker cannot RE the inputs to the mix operation in a sequencing graph, as shown in Fig. 5(b).*

The mixer in Fig. 5(a) has a ring with one inlet channel $A \rightarrow B$ and an outlet channel $E \rightarrow O$. The mixing time can be deduced from a sequence of opening and closing of the inlet/outlet channels followed by the peristaltic pumping operation. The status of the channels $A \rightarrow B$ and $E \rightarrow O$ can be obfuscated by adding extra valves; viz Equation (3). This leads to ambiguity in the mixing time and the number of mixing steps. The following example describes it in detail.

Example 2. *In Fig. 5(a), consider the valve types as in Example 1, i.e., all* $g_* = 1$ *except* $g_{1b}, g_{2b}, g_{3b}, g_{10b} = 0000$*. Let the valve actuation be* $v_{1a}, v_{1b}, v_{2a}, \cdots, v_{10a}, v_{10b}, v_{11} = 011010101111100$*. This opens the mixer inlet/outlet to push out the mixer content which denotes the end of the previous mixing step. If the actuation is followed by a peristaltic*

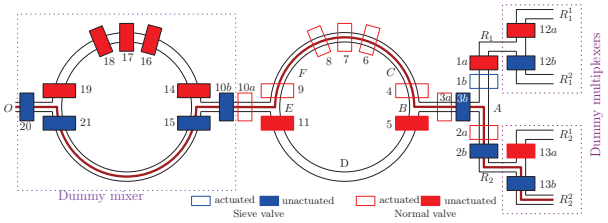

Fig. 6: Structural obfuscation: a dummy multiplexer addition at inlets R_1, R_2 and a dummy mixer addition at outlet O.

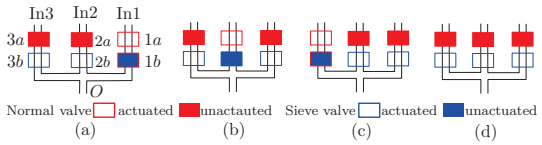

Fig. 7: Multiplexer actuation to push fluid through (a) In1, (b) In2, (c) In3, and (d) no fluid.

pumping operation, then it denotes the start of a new mixing. On the other hand, if $\{3a, 10a\}$ *are sieve valves and* $\{3b, 10b\}$ *are normal valves, then the given actuation set does not open the inlet/outlet of the mixer. Hence, the previous mixing step has not ended and a new mixing step has not started. This leads to obfuscation in the deduction of the mixing time and the number of mixing steps, as shown by the dotted nodes in the sequencing graph in Fig. 5(b).*

D. Structural Obfuscation

The *behavioral obfuscation* does not change the structure of the biochip but inserts extra valves on the existing channels. Furthermore, the structure of the biochip can be obfuscated by inserting dummy channels, multiplexers and/or mixers. This is *structural obfuscation*. A channel can be mimicked by a dummy multiplexer with a sieve valve on the original inlet - so that it is always open and a normal valve on a dummy inlet - which is kept closed. Without the knowledge of sieve valve, the attacker cannot know which inlet is selected when both the valves are closed. Alternately, the channel can be mimicked by a dummy mixer with sieve valves forming an always open channel in the ring mixer. The valves of this module are actuated like a mixing module to mislead the attacker. To resolve this ambiguity, an attacker has to do trial-and-error by replacing each mixing operation in the actuation with a transportation operation.

Example 3. *In Fig. 6,* $12b, 13b$ *are sieve valves and* $12a, 13a$ *are normal valves. For actuation set* $v_{12a}, v_{12b}, v_{13a}, v_{13b} = 0000$*, paths* $R_1^2 \rightarrow R_1$ *and* $R_2^2 \rightarrow R_2$ *are open. On the other hand, if* $12b, 13b$ *are normal valves and* $12a, 13a$ *are sieve valves, then for the same actuation set* $R_1^1 \rightarrow R_1$ *and* $R_2^1 \rightarrow R_2$ *are open. This leads to obfuscation of the fluid selected. Furthermore, a dummy mixer with* $\{15, 20, 21\}$ *as sieve valves is added to path* $E \rightarrow O$*. The valves of this mixer can be actuated to mimic a normal mixer, whereas in reality, it is a* $E \rightarrow O$ *channel controlled by valve* $10a$*.*

V. Design for Obfuscation

We define the security metrics that capture the security-cost trade-offs and design-for-obfuscation rules.

A. Security Metrics

To RE the bioassay, the attacker has to interpret the actuation sequences that are ambiguous due to unknown valve type g_*. Such actuations are referred to as *ambiguous actuations*. The attacker can build a biochip prototype from the snapshots without correct valve types. By trial-and-error, the attacker can replace the ambiguous actuations until the results of the bioassay on the biochip prototype become identical to the known sensor readings. The maximum number of experiments required to resolve this ambiguity is denoted as *resolution effort* $\mathcal{E}$. The biochip designer obfuscates the biochip and its actuation sequence to make RE hard-enough to deter an attacker. The design overhead for obfuscation is defined in terms of extra valves, which in turn may lead to extra pins in the biochip and extra memory for storing the corresponding actuation signals. To maximize the resolution effort, we propose the following design rules.

B. Design-for-Obfuscation Rules

In a crude attack, the attacker will try all combinations of g_*. However, a smart attacker will leverage functional properties to prune the search space. To achieve a robust obfuscated design, we frame four design rules.

1) Channel: To push a fluid in a CFMB, an input-output channel needs to be opened. The attacker tries to identify which input-output path is opened in a given cycle. If there exists an input-output path without any unactuated valve, then the actuation is unambiguous to the attacker. Else, the attacker has to guess if any of the unactuated valves on the input to output paths is a sieve valve. This leads to the first design rule.

> Rule #1: In an ambiguous actuation, every input to output channel path must have at least one closed sieve valve.

Consider a channel that has n_{chl} unactuated valves in an ambiguous actuation cycle. Without knowing the valve type g_* i.e., sieve or normal, the de-obfuscation effort $\mathcal{E}_{chl}$ involves trials that map each closed valve to two possibilities - closed and open. Hence, $\mathcal{E}_{chl} \leq 2^{n_{chl}}$. The effort increases with the number of distinct input-output paths with closed valves in a cycle.

2) Multiplexer: An attacker can use the following properties of a multiplexer to resolve the obfuscation. *P1*: At most one path of the multiplexer can be open at any time. *P2*: It is likely that each inlet fluid is selected at least once in a bioassay. An attacker can collect all the unique actuations applied to the multiplexer and along with the properties *P1* and *P2* the attacker can de-obfuscate the multiplexer actuations as discussed in the following example.

Example 4. *Consider a 3-inlet multiplexer with two valves* $*a$ *and* $*b$ *on each inlet. For each inlet, the set of actuations* $v_{*a}, v_{*b} = \{11, 00\}$ *is unambiguous and* $v_{*a}, v_{*b} = \{10, 01\}$ *is ambiguous. Between any pair of inlets, there are four possible combinations of these ambiguous actuations. In Fig. 7, 3-out-of-4 combinations are used for actuating the valves* $v_{1a}, v_{1b}, v_{2a}, v_{2b}$*. The unused ambiguous actuation combination on the inlet* In1 *and* In2 *is* $v_{1a}, v_{1b}, v_{2a}, v_{2b} = 1010$*. The attacker can decipher that this actuation opens both inlets* In1 *and* In2 *and hence is not used due to property P1. Alternately, an attacker can guess that the least used actuation on an inlet is used to open the respective inlet. In Fig. 7, actuation*

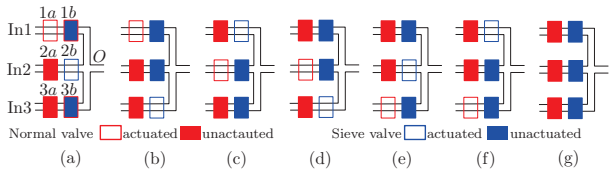

Fig. 8: Multiplexer actuation for (a-b) pushing In1, (c-d) pushing In2, (e-f) pushing In3 and (g) not pushing any fluid.

$v_{*a}, v_{*b} = 10$ is the least used actuation on each inlet. The attacker can decipher with high probability that these actuations open their respective inlets due to property P2.

A naive defense against the above attacks is to increase the number of valves on each inlet. However, this increases the cost. To avoid cost escalation, we use two valves per inlet with design rules #2 and #3.

Rule #2: Apply ambiguous actuations to no more than two inlets at a time. One inlet is the fluid being pushed and one from the other $m-1$ inlets of the multiplexer.

Rule #3: Apply unambiguous actuations when no fluid is pushed through the multiplexer.

In an m-inlet multiplexer, through these design rules, there are $m-1$ ways of actuating an obfuscated push operation of a fluid. The ambiguous actuation on the same inlet can be used in the obfuscated push operation of other $m-1$ inlets. This defeats the two attacks described in Example 4. The maximum number of unique ambiguous actuations is $s_{mux} = \binom{m}{2}$, as shown in Fig. 8. Each ambiguous actuation can be mapped to two possibilities. If there are s unique ambiguous operations in a bioassay, an attacker needs to perform 2^s experiments. The maximum resolution effort is $\mathcal{E}_{mux} = 2^{s_{mux}} = 2^{\binom{m}{2}}$.

3) Mixer: Reliable mixing requires a minimum mixing time, which depends on the frequency of actuations, channel geometries etc. If ambiguous actuations are inserted in the mixer actuation sequence prior to the minimum mixing interval, then the attacker can map that actuation to an ongoing mix operation and prune the search space. To avoid this, we frame design rule #4.

Rule #4: The gap between an ambiguous and other mix operations must be more than the minimum mix time, t_{min}.

The number of ambiguous mixer actuations is dependent on the number of valves on the mixer inlet and outlet, provided rule #4 is satisfied. However, to minimize the cost we use two valves on the mixer inlet (outlet). The number of possible ambiguous actuations on the mixer inlet (outlet) is two. This implies that the maximum number of ambiguous actuations that can be applied to the mixer (inlet and outlet) is $s_{mix} = 4$. The ambiguous actuations can be mapped to one of the two possibilities - a new-mix operation or no new-mix operation. Therefore, the RE effort for a mixer is $\mathcal{E}_{mix} = 2^{s_{mix}} = 2^4$.

4) Dummy Structures: The same rules apply to dummy structures such as multiplexers and mixers (Fig. 6). To resolve the ambiguity about n_{dum} dummy structures, ($\mathcal{E}_{dum} = 2^{n_{dum}}$) trial experiments must be performed. However, the cost of introducing dummy structures include not only extra valves but also extra channels and extra input/output ports.

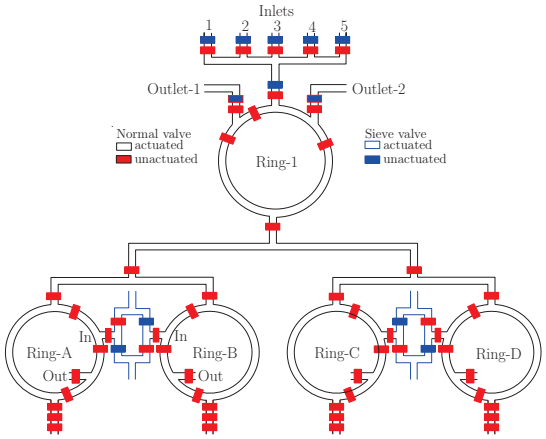

Fig. 9: AutoChIP used for gene enrichment. The extra flow channels are shown in blue color.

VI. Experimental Results

We analyze the application of the obfuscation techniques on a chromatin immunoprecipitation (ChIP) biochip. Then, we obfuscate other real-life biochips.

A. Chromatin Immunoprecipitation (ChIP)

The ChIP performs a two-step bioassay: 1) Cell lysis and DNA fragmentation are performed on the sample cells through a series of mixing operations (Fig. 9). This step uses a 5:1 multiplexer that selects cells and reagents being pushed into the mixer Ring-1. 2) The resulting fluid is divided equally into four rings (A-D) to perform ImmunoPrecipitation. These rings are preloaded with anti-body functionalized beads and mixed with cellular material from step 1. Next, contents of the each of the four rings (A-D) is washed with four different wash buffers (Fig. 10(a)). The washed beads are then moved to micro-centrifuge tubes for qPCR analysis [13].

We apply behavioral obfuscation to step 1 and structural obfuscation to step 2 as follows: One extra valve is added to each inlet of the 5:1 multiplexer, the ring-1's inlet, and two outlets, i.e., a total of eight extra valves are used, as shown in Fig. 9. This obfuscates the multiplexer selection, the number of mixing operations, and the mixing time. The maximum number of ambiguous actuations applied to the multiplexer and mixer are $s_{mux} = \binom{5}{2}$ and $s_{mix} = 4$, respectively. An attacker's effort in resolving the behavioral obfuscation is $\mathcal{E}_{behav} = \mathcal{E}_{mux} \cdot \mathcal{E}_{mix} = 2^{\binom{5}{2}} \cdot 2^4 = 2^{14}$.

The four ring mixers (A-D) are connected to four fluid inlets that are used to wash the contents of the respective mixer. The inlet channel is replaced by a dummy multiplexer to select between the original wash fluid and a wash fluid corresponding to other mixers, as shown in Fig. 9. This results in eight more valves and four more channels. The effort to resolve the structural obfuscation of $n_{dum} = 4$ multiplexers is $\mathcal{E}_{struct} = 2^{n_{dum}} = 2^4$. The obfuscated sequencing graph is shown in Fig. 10(b). The effort to resolve the behavioral + structural obfuscation is $\mathcal{E} = \mathcal{E}_{behav} \cdot \mathcal{E}_{struct} = 2^{14} \cdot 2^4 = 2^{18}$. Each ChIP trial takes 3.5 h. The time for all trials is over a thousand years. Also, each trial consumes reagents, samples, and biochips.

TABLE II: Obfuscation of real-life biochips.

Biochip	# Mux (m:1)	# Ring mixers	# Other channels	# Valves	Behavioral obfuscation		Structural obfuscation				Total effort $\mathcal{E}_{struct} \cdot \mathcal{E}_{behav} = \mathcal{E}$
					# Extra Valves	Effort	# Extra channels	# Extra valves	#Extra inputs	Effort	
ChIP [13]	1 (5:1)	4	1	50	8	2^{14}	4	8	0	2^4	$2^{14} \cdot 2^4 = 2^{18}$
Kinase act [18]	2 (3:1)	2	2	44	5	2^7	3	6	3	2^3	$2^7 \cdot 2^3 = 2^{10}$
mRNA iso. [19]	4 (2:1)	4	1	56	16	2^6	4	8	4	2^4	$2^6 \cdot 2^4 = 2^{10}$
Nucleic-Acid proc. [2]	3 (5:1)	3	3	54	12	2^{13}	3	6	0	2^3	$2^{13} \cdot 2^3 = 2^{16}$

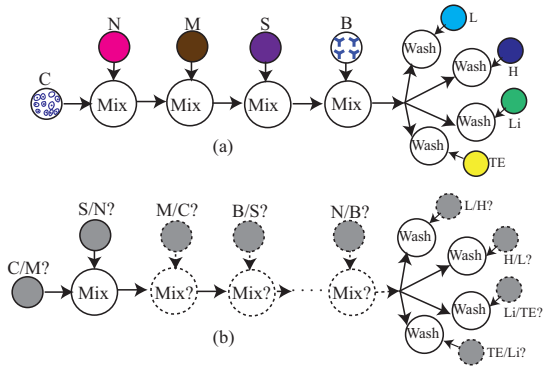

Fig. 10: ChIP bioassay: (a) original and (b) obfuscated. C: Cells under test, N: NP40 buffer, M: Microccal nuclease enzyme, S: SDS/EDTA lysis buffer, B: antibody fictionalized beads, L- Low salt buffer, H- High salt buffer, Li- LiCl buffer, and TE- TE buffer.

B. Other Benchmarks

We applied sieve-valve-based obfuscation to three more real-life biochips and tabulated the results in Table II. The mRNA iso. and Kinase act. are 4-plex and 2-plex biochips, respectively, where identical assays (attacker trials) are run in parallel. In mRNA iso. ($4 \times 14 = 56$ valves) and Kinase act. ($2 \times 22 = 44$ valves) biochips, $\mathcal{E}$ is smaller due to replication of a smaller structure. On the other hand, in the larger biochips like ChIP (50 valves) and Nucleic-Acid proc. (54 valves), $\mathcal{E}$ is larger for a comparable design cost in terms of the number of extra valves. The results imply that the sieve-valve-based obfuscation scales well with the complexity of the biochip. The extra valves and inputs indicate the obfuscation cost.

VII. Conclusion

Microfluidic platforms have immense potential in paving the way for rapid and low-cost biochemical analysis. However, the cyberphysical system that enables biochip automation is susceptible to IP theft. This is a major hurdle in the large-scale adaptation of microfluidic technologies in industries that are prone to stealing of sensitive research data. Our work addresses this pressing problem with a practical obfuscation methodology that can be easily integrated with the current biochip design flow. We developed sieve-valve-based obfuscation design rules and showcased their application to the real-life biochips. The results show that the de-obfuscation effort is daunting enough to act as a deterrent to an attacker.

The strength of our proposal can be demonstrated in comparison with two IP protection techniques. First, firmware encryption has been used to protect firmware IPs. However, this doesn't apply to the biochips because the biochip actuations are electrical signals applied to either the valves or to the pneumatic actuators. Even if the actuation sequence is encrypted, it has to be decrypted before it is applied to the biochip control ports. Further, the actuations can be extracted by image and video-based RE. Proposed obfuscation complements encryption to thwart RE of the electrical signals. Second, logic locking is used to prevent IP piracy in VLSI designs. The number of trials needed to de-obfuscate a logic-locked VLSI design is of the order of 2^{128} [20]. These trials can be done on high-speed computers. On the other hand, the bioassay trials take several hours to complete. Also, unlike VLSI, the bioassay recovery trials require perishable reagents and biochips. The cost and time spent on these trials go against to an attacker's economic objective of stealing a bioassay IP.

Acknowledgment

This research is supported in part by the Army Research Office under grant number W911NF-17-1-0320, NSF Award numbers CNS-1833622 and CNS-1833624, NYU Center for Cyber Security (CCS), and CCS-AD.

References

[1] A. H. C. Ng *et al.*, "Immunoassays in microfluidic systems," *Anal. Bioanal.Chem.*, vol. 397, no. 3, pp. 991–1007, Jun 2010.
[2] J. W. Hong *et al.*, "A nanoliter-scale nucleic acid processor with parallel architecture," *Nature Biotechnology*, vol. 22, p. 435439, 2004.
[3] Y.-H. V. Ma *et al.*, "A review of microfluidic approaches for investigating cancer extravasation during metastasis," *Microsystems & Nanoengineering*, vol. 4, no. 17104, pp. 1–13, 2018.
[4] (2018) MARKETandMARKET. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/microfluidics-market-1305.html
[5] S. S. Ali *et al.*, "Microfluidic encryption of on-chip biochemical assays," in *Proc. IEEE Biomed. Circuits Syst. Conf.*, 2016, pp. 152–155.
[6] (2016) Drug development and intellectual property theft. [Online]. Available: https://digitalguardian.com/blog/drug-development-and-intellectual-property-theft
[7] (2016) 2 GSK scientists indicted in secrets case involving China. [Online]. Available: https://www.justice.gov/usao-edpa/pr/scientists-indicted-allegedly-stealing-biopharmaceutical-trade-secrets
[8] P. S. Dittrich *et al.*, "Lab-on-a-chip: microfluidics in drug discovery," *Nature Reviews Drug Discovery*, vol. 5, no. 3, pp. 210–218, 2006.
[9] J. Melin *et al.*, "Microfluidic large-scale integration: The evolution of design rules for biological automation," *Annual Review of Biophysics and Biomolecular Structure*, vol. 36, no. 1, pp. 213–231, 2007.
[10] H. Chen *et al.*, "Biochipwork: Reverse engineering of microfluidic biochips," in *Proc. IEEE Intl Conf. on Computer Design*, 2017, pp. 9–16.
[11] J. P. Urbanski *et al.*, "Digital microfluidics using soft lithography," *Lab Chip*, vol. 6, pp. 96–104, 2006.
[12] S. R. Quake *et al.*, "Microfluidic sieve valves," Jan 2015, US Patent 8932461B2. [Online]. Available: https://patents.google.com/patent/US8932461B2/en
[13] A. R. Wu *et al.*, "Automated microfluidic chromatin immunoprecipitation from 2,000 cells," *Lab Chip*, vol. 9, pp. 1365–1370, 2009.
[14] S. Bhattacharjee *et al.*, "Locking of biochemical assays for digital microfluidic biochips," in *IEEE European Test Symp.*, 2018, pp. 1–6.
[15] (2018) Laboratory monitoring. [Online]. Available: http://tetrascience.com/case-studies/laboratory-monitoring-notable-labs
[16] U. D. of Homeland Security *et al.*, "Russian government cyber activity targeting energy and other critical infrastructure sectors," 2018.
[17] R. Langner, "Stuxnet: dissecting a cyberwarfare weapon," *IEEE Security Privacy*, vol. 9, no. 3, pp. 49–51, 2011.
[18] C. Fang *et al.*, "Integrated microfluidic and imaging platform for a kinase activity radioassay to analyze minute patient cancer samples," *Cancer Research*, vol. 70, no. 21, pp. 8299–8304, 2010.
[19] J. S. Marcus *et al.*, "Microfluidic single-cell mRNA isolation and analysis," *Analytical Chemistry*, vol. 78, no. 9, pp. 3084–3089, 2006.
[20] J. Rajendran *et al.*, "Fault analysis-based logic encryption," *IEEE Trans. on Computers*, vol. 64, pp. 410 – 424, 2015.